Data Aggregation with Principal Component Analysis in Big Data Wireless Sensor Networks

Jun Li*, Songtao Guo*, Yuanyuan Yang† and Jing He*
*College of Electronic and Information Engineering, Southwest University, Chongqing 400715, P. R. China
† Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794, USA

***Abstract*—In wireless sensor networks (WSNs), numerous sensors can produce a significant portion of the big data. It remains an open issue how to timely gather and transmit such large amount of data while minimizing data latency through wireless sensor networks (WSNs). On the other hand, spatially correlated sensor observations lead to considerable data redundancy in the network. To efficiently eliminate data redundancy and improve energy efficiency, in this paper, based on the fact that the more similar the measure data are, the smaller the amount of data after aggregation is, we first develop a new distributed clustering algorithm which can categorize sensor nodes with high similarity into a cluster for data aggregation, while ensuring uniform energy consumption within the cluster. Then, we propose a data aggregation algorithm based on principal component analysis (PCA) which can be executed in the cluster head (CH). Finally, our experimental results demonstrate that the amount of data transmission can be significantly reduced based on our proposed clustering and data aggregation algorithm.**

***Index Terms*—Data aggregation, big data, wireless sensor networks, principal component analysis.**

I. Introduction

In wireless sensor networks (WSNs), numerous sensors can produce a significant portion of the big data when they are used extensively in many military and civilian applications, such as target tracking, environment monitoring, health monitoring, and observing phenomena [1]. Usually, the big data contain high volume, high velocity, and high variety information assets, which are difficult to collect, process, and transmit by using existing algorithms and models. In such applications, if the sink node wants to know the real-time information in the sensing region, data transmission with low latency is very crucial [2]. However, how to transmit such large amount of data while minimizing data latency still remains a challenging issue in wireless sensor networks [3].

A number of compression methods such as compressed sensing (CS) [4], principal component analysis (PCA) and wavelet compressive [5] have been proposed. Wavelet compressive is a powerful tool for non-stationary signal. In data transmission, instead of sending raw sensing data, the way of wavelet transform transmits the wavelet coefficients to the sink node. At last, the sink node can recover the full raw data set by an inversion transformation. In practical application, there are a variety of wavelet bases need to be chosen. However, once the wavelet base is selected, its characteristic is fixed, and it is difficult to accurately approximate the local signal characteristics at different scales. Compressed sensing (CS) is a complex compression method. CS projects the high dimension into low dimension space to reduce the transmission of data. CS can achieve a precise recovery with fewer measurements than the dimension of the raw data. The complexity of the encoding process of CS is very low, thus CS is often used in data aggregation [6]. However, the challenge of CS is decoding computation. The decoding with high complexity not only requires high computing capacity, but also causes longer processing delay. In the case of applying CS into large-scale WSNs, the excessive requirements for computing capacity and processing delay will greatly restrict the application of CS. Compared with CS, principal component analysis (PCA) can be executed in the cluster head (CH), which can quickly process the data, and be used in exchange for the most effective data aggregation of a minimal energy and delay. Thus PCA would be a promising solution in large scale WSNs.

On the other hand, the sensing regions among sensor nodes in WSNs are usually overlapped and dependent spatially, which makes the observed data have a certain spatial correlation [7] [8]. Therefore, spatially correlated sensor observations lead to considerable data redundancy in the network. To efficiently reduce data latency and increase energy efficiency in data transmission, it is highly desirable to eliminate such data redundancy through effective data aggregation. To this end, researchers have proposed various data aggregation approaches. LEACH is a typical clustering protocol [9]. It achieves energy saving by changing the structure of the network, however, it is pre-selected cluster head node and CHs are fixed until the end of the life cycle of the network. In [10], Wang et al designed the single-hop-length (SHL) and multiple-hop-length (MHL) schemes for optimal aggregation throughput, and considered the tradeoff between aggregation throughput and gathering efficiency. In [11], Hua et al presented an optimal routing and data aggregation scheme by exploiting the special structure of the sensor network. In [12], Barton et al showed that data aggregation rate of $\Theta(\log n/n)$ per node is optimal. Liu et al [13] explored temporal correlation in each cluster, but they only send a part of the sensed data to the sink node, which cannot accurately represent the specific information in the network. In [14], the authors proposed a clustering approximation framework based on a grid-based spatial correlation clustering method which clusters the sensor nodes according to data correlation. It can really reduce the transmission of data at the cost of data

978-1-5090-5696-5/16 $31.00 © 2016 IEEE
DOI 10.1109/MSN.2016.14

accuracy. However, the aforementioned algorithms ignore the characteristics of data themselves.

In this paper, to efficiently transmit the sensed big data with low latency, we propose a principal component analysis (PCA) based data aggregation algorithm to eliminate data redundancy in cluster head nodes so as to minimize the amount of data transmitted. Clearly, the more similar the sensed data of nodes are, the smaller the amount of data after aggregation is for a given normalized reconstruction error in PCA. Based on this consideration, therefore, we give a definition of data similarity suitable for principal component analysis. Compared to existing aggregation algorithms, which focuses on locations of sensors, our algorithm pays more attention to the similarity of data themselves.

The contributions of this paper can be summarized as follows.

- Firstly, in order to balance the energy consumption among clusters and avoid data conflicts within a cluster, we find the optimal number of members in each cluster.
- Moreover, we propose a distributed clustering algorithm based on the similarity to put the nodes with high similarity into the same cluster.
- Furthermore, on the basis of the constructed cluster, we propose the data aggregation algorithm based on PCA on cluster head nodes to deal with the similar data from the cluster members.
- Finally, we verify by simulation that our algorithm can efficiently minimize the amount of data transmission on cluster heads while ensuring low latency of data transmission.

The rest of this paper is organized as follows. Section II analyzes the optimal number of members in a cluster by an energy model. Section III proposes a clustering algorithm based on data similarity and Section IV proposes the data aggregation algorithm based on PCA. Section V evaluates the performance of the proposed algorithm. Finally, Section VI concludes this paper.

II. System Energy Model

A large number of sensors in wireless sensor networks will produce a large amount of data. These sensed data are gathered into big data [15]. When sink node wants to spend the least energy to obtain all of the information, we have to first understand the distribution of the big data. There are two ways to get the distribution of the data. One is that the sensor node processes and extracts the related data, then transmits the processed data to sink node. The other is that when transmitting the big data, all nodes transmit their raw data to the sink node. The sink node deals with these data, and makes the best choice for the network. In this paper, we adopt the second approach to find the data of the neighbor nodes with high similarity, and put the data together to process with the data compression of the PCA.

We consider a sensor network consisting of N sensor nodes that are randomly distributed in an $M \times M$ sensing area. We assume that each node has l bytes of data to be sent to the sink node, and a simple model for the radio transmission of energy consumption. The energy consumed by transmitting l bytes in each node is given by [14]

$$E_T(l,d) = \begin{cases} l * E_{elec} + l * E_{fs} * d^2 & , if\ d < d_0 \\ l * E_{elec} + l * E_{amp} * d^4 & , \text{otherwise,} \end{cases} \tag{1}$$

where d is the distance between transmitter and receiver and d_0 represents the communication radius of the node. E_{elec} is energy consumption of the circuit per byte by the transmitter and the receiver, and $E_{fs} * d^2$ is the energy consumption of the amplifier in the communication range. Within the communication range of a node, it maintains a certain power operation. When the communication distance is beyond the communication scope of sensor node, the node needs to increase the transmission power. $E_{amp} * d^4$ denotes the energy consumption of the amplifier beyond the communication range.

Moreover, we divide the network into clusters based on data similarity of nodes, and non-cluster head node transmits the data to the cluster head. Let E_{pr} be the energy consumed by the cluster head aggregating one byte data from its cluster members. Then we have

$$E_P = k * l * E_{pr}, \tag{2}$$

where E_P represents energy consumption of data aggregation in a cluster head, and k is the number of cluster members. Clearly E_P is proportional to l. In data reception, the energy consumption for receiving l bytes of data for each node can be given by

$$E_R(l,d) = l * E_{elec}. \tag{3}$$

Energy consumption sources of each cluster includes the data transmission, data reception and data processing section. Therefore, we can easily formulate the energy consumption in the cluster as

$$\begin{aligned} E_{cluster} &= E_T(l,d) + E_R(l,d) + E_P \\ &\approx k * l * E_{elec} + k * l * E_{fs} * d_{toCH}^2 \\ &+ \beta * l * E_{elec} + \beta * l * E_{amp} * d_{toSINK}^4 \\ &+ k * l * E_{elec} + k * l * E_{pr} \\ &= 2k * l * E_{elec} + k * l * E_{fs} * d_{toCH}^2 \\ &+ \beta * l * E_{elec} + \beta * l * E_{amp} * d_{toSINK}^4 \\ &+ k * l * E_{pr}, \end{aligned} \tag{4}$$

where d_{toCH} and d_{toSINK} denote the distance from the node to the cluster head and the sink node, respectively. We assume that the data aggregation is perfect aggregation. $\beta * l$ denotes the size of data after aggregation. According to [16], the expected value of squared distance d_{toCH}^2 can be denoted by

$$E(d_{toCH}^2) = \frac{1}{2\pi}\frac{M^2}{N/k} = \frac{k}{2\pi}\frac{M^2}{N}. \tag{5}$$

Moreover, the total energy consumption for one round of data collection is given by

$$\begin{aligned} E_{total} &= \frac{N}{k} * E_{CH} \\ &= 2N * l * E_{elec} + N * l * E_{fs} * \frac{k}{2\pi}\frac{M^2}{N} \\ &+ \frac{N}{k} * \beta * l * E_{elec} + \frac{N}{k} * \beta * l * E_{amp} * d_{toSINK}^4 \\ &+ N * l * E_{pr}. \end{aligned} \tag{6}$$

Then we can obtain the optimal number of members in a cluster by taking the first-order derivative of E_{total} with respect to k and letting it be zero, i.e.,

$$k_{opt} = \sqrt{\frac{2\pi(N\beta E_{elec} + N\beta E_{amp} d_{toSINK}^4)}{E_{fs} M^2}}. \tag{7}$$

It is not difficult to observe that the optimal number of members in a cluster is related to the total number of sensor nodes in the network, the distance between the sensor node and sink node, and the size of the sensing region. k_{opt} is the theoretically optimal value and we will use specific simulations to validate the value.

III. Clustering Algorithm with Data Similarity

A. Similarity Measurement

In order to ensure the accuracy, we often need the real-time data transmission. Due to high spatiotemporal correlation, the sensed data by different sensor nodes are similar. Thus, we will propose a clustering algorithm based on the data similarity for data aggregation. In the following, we define the data similarity from two aspects: data magnitude similarity and data correlation.

1) Data magnitude similarity: We assume that D denotes the difference of current data of nodes, and c denotes a measurement threshold. We can set the appropriate value of c according to the actual requirements, i.e.,

$$c = D \times kk, \tag{8}$$

where kk denotes the similarity coefficient, which reflects the influence magnitude of the difference of data on the similarity. We can change the values of kk by need. According to the characteristics of our collected data, we let $kk = 0.3$. If the measured values of the sensed data from two neighboring nodes are less than c, their data meet the magnitude similarity. Within one hop of data transmission, it is easy to prove that the difference of the node measurements in the same cluster is less than $2 * c$.

2) Data correlation: We assume that $X_n = \{x_{n1}, x_{n2}, \ldots, x_{nt}\}$ represents the observation of node n, where the series $1, 2, \ldots, t$ is a sampling time slots. And we assume that nodes 1 and 2 are neighbors and their observations can be represented by $X_1 = \{x_{11}, x_{12}, \ldots, x_{1t}\}$ and $X_2 = \{x_{21}, x_{22}, \ldots, x_{2t}\}$, respectively. Data correlation between nodes 1 and 2 can be defined as

$$corr(X_1, X_2) = \frac{L_{12}(X_1, X_2)}{\sqrt{L_1(X_1) \times L_2(X_2)}}, \tag{9}$$

where

$$L_{12}(X_1, X_2) = \sum_{i=1}^{n}(x_{1,i} - \overline{X_1})(x_{2,i} - \overline{X_2})$$

$$L_1(X_1) = \sum_{i=1}^{n}(x_{1,i} - \overline{X_1})^2, L_2(X_2) = \sum_{i=1}^{n}(x_{2,i} - \overline{X_2})^2$$

$$\overline{X_1} = \frac{1}{n}\sum_{i=1}^{n} x_1, \overline{X_2} = \frac{1}{n}\sum_{i=1}^{n} x_2.$$

We assume that X_1 and X_2 are similar if they can meet the following metric

$$corr(X_1, X_2) \leq \varepsilon. \tag{10}$$

If the similarity measure of a sensor node satisfies the data magnitude similarity and data correlation, i.e., the similarity measure is below a given threshold c and ε, we say they are similar.

B. Clustering-compression algorithm

In this subsection, given the similarity values, we propose a heuristic clustering algorithm suitable for the PCA based aggregation algorithm to partition the sensor nodes with high similarity into clusters, and select an appropriate sensor as the cluster head (CH) in a cluster.

Our heuristic clustering algorithm can be described as follows. Each sensor node perceives environmental data at a fixed interval, and the perceived data constitute the observation metric X. First, each node will calculates the similarity with its neighbor nodes. If node u and node v satisfy a given similarity threshold ε, then they will set up an edge uv. Such that, all sensor nodes can form a graph G. Then, we will sort nodes by their degrees, and the node with the largest degree is selected as the cluster head. To minimize energy consumption in each cluster, we constrain the number of members in each cluster to $k-1$. The value of k is equal to k_{opt} in Eq. (7). Then, the cluster head (CH) will select $k-1$ nodes with higher similarity from its neighbor nodes, and we remove the selected nodes from the set S of all the nodes. The clustering procedure repeats until the largest degree of nodes in S is less than $k-1$. In general, there remain some nodes in S. In this situation, we will not activate the computation of similarity (c and ε), but reduce the number of cluster members so as to make the remaining node become a cluster. Once clusters become stable, cluster members will make the cluster head rotation. When the sink node finds that intercluster data have greater difference than a given threshold a few times or half of the nodes do not satisfy the similarity values, it will decide to renew to activate the clustering algorithm.

Algorithm 1 Similarity Clustering Algorithm

Input: N nodes in set S

Output: Clusters with k members and clusters with less than k members;

1: Each node records the number of its neighbor nodes;
2: **while** $k > 1$ & $S = \emptyset$ **do**
3: **for** $j = 1 \rightarrow n$ **do**
4: Node j in S measures the similarity based on our proposed criteria, i.e., the difference of the measurement with its neighbor nodes is less than c, and corr(X_j, $X_{j'neighbornode}$) is less than ϵ;
5: Connect node j with its neighbor nodes which satisfy the given similarity threshold c and ϵ;
6: Calculate the degree of node j;
7: **end for**
8: **repeat**
9: Rank the nodes according to their degrees in S;
10: Select the node with largest degree as the CH node;
11: Successively select $k-1$ neighbor nodes with the larger similarity value as cluster members;
12: Remove the CH node and its cluster members from S;
13: Update the degree of each remaining node in S.
14: **until** the largest degree falls below $k-1$
15: $k = k - 1$;
16: Let n equal to the number of elements in the current set S.
17: **end while**

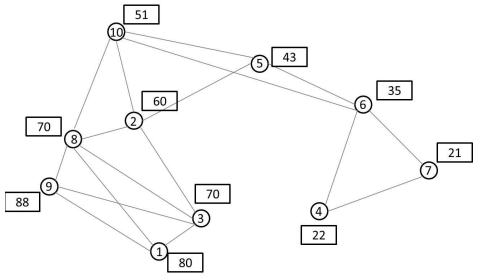

Fig. 1. Similarity measurement.

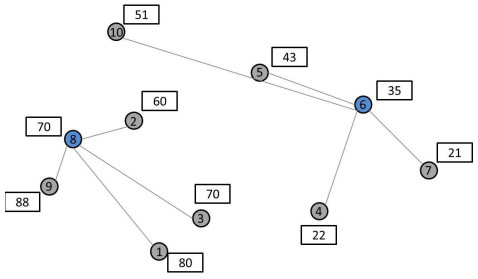

Fig. 2. Clustering with $d = D * 0.3$.

The pseudo-code procedure of the algorithm is given in Algorithm 1.

Fig. 1 shows the results of the similarity measurement while Fig. 2 shows the results of the proposed clustering algorithm 1, where we set $c = 20$ and $\epsilon = 0.8$, the number in the circle denotes the number of nodes, the number outside the circle indicates the current measured data, and the source of data can be found in the experimental Section V. We can observe from Fig. 2 that the similarity measurement of adjacent nodes are similar in the same cluster after clustering.

IV. Data Aggregation based on Principal Component Analysis

In this section, we propose a data aggregation algorithm based on principal component analysis.

Principal component analysis (PCA) is a useful compression algorithm, and is quite suitable for data aggregation in sensor nodes with limited computation capacity in WSNs [17]. PCA transforms the sensed data into a new coordinate system and makes eigenvector of the maximum eigenvalue become the first coordinate (called the first principal component), the second one become the second coordinate (called the second principal component), and so on [18]. Therefore, PCA can reduce the dimension degree of data sets while keeping the characteristics of the largest contribution to the variance.

In the following, we introduce the data compression method based on PCA. A wireless sensor network can be divided into many clusters by our proposed clustering method. Cluster head (CH) will collect measurement data of its members and then put the measurement data into the observation matrix x. The observation matrix x can be transformed into a new space by

$$y = Px, \tag{11}$$

where P denotes an $m \times n$ orthogonal transformation matrix, x indicates an $n \times n$ matrix, and m is much smaller than n. Then we can obtain a low-dimension projection matrix y. If we would like to find a suitable matrix P, x can be reconstructed based on the following equation

$$\overline{x} = P^T y. \tag{12}$$

Since the cluster head node only sends y to its destination, instead of the high-dimension x, the amount of the transmitted data can be reduced remarkably, which will further reduce the energy consumption of data transmission [19]. Because the dimension of y is reduced, and $\overline{x}$ is only an approximation of x, we consider the normalized reconstruction error defined as

$$\gamma = \frac{\| x - \overline{x} \|_2}{\| x \|_2}, \tag{13}$$

where the vectors x and $\overline{x}$ represent the original and recovered data, respectively.

It is clear that the reconstruction error γ is determined by transformation matrix P. The matrix P comes from covariance matrix C, and each column vector of matrix P is the eigenvector of C [20]

$$C = E[(x - E[x])(x - E[x])^T]. \quad (14)$$

Let λ indicate the eigenvalue vector of the covariance matrix C, where the eigenvalues are nonnegative real numbers since any covariance matrix is nonnegative definite [21]. q denotes the number of non-zero eigenvalues of C ($q \geq m$). If all q eigenvectors of matrix C are used as the columns of P, the normalized reconstruction error will be minimized, and total variances of data in all directions will be saved, and y will represent all the principal components of x. In other words, in order to ensure the accuracy of data reconstruction, we must limit the size of projection space. The CH will use the following formula to measure the accuracy of data reconstruction

$$T(m) = \frac{\sum_{k=1}^{m} \lambda_k}{\sum_{k=1}^{n} \lambda_k}. \quad (15)$$

where λ_k denotes the k-th largest eigenvalue. The pseudo-code procedure of the data aggregation algorithm is given in Algorithm 2. We let d_{value} denote the difference between the maximum and minimum elements of the observation matrix X

$$d_{value} = X_{max} - X_{min}, \quad (16)$$

where X_{max} and X_{min} denote the maximum and minimum elements in X, respectively. We let ρ denote the aggregation ratio, which is the ratio of the size of data after aggregation to that before aggregation. Fig. 3 depicts the relationship between the normalized reconstruction error and the aggregation ratio for different d_{value}. It is not difficult to find from Fig. 3 that (i) the normalized reconstruction error decreases as the aggregation ratio increases; (ii) the smaller the value of d_{value} is, the smaller the aggregation ratio is for a given reconstruction error. When the error rate is 5%, the aggregation ratio of $d_{value} = 50$ is 22%. However, the aggregation ratio of $d_{value} = 150$ is 44% and the aggregation ratio of $d_{value} = 100$ is 40%. In this case, it will save around half storage space. Thus the data similarity has a great influence on the data compressed.

V. Performance Evaluation

In this section, we present numerical results to verify the performance of our proposed clustering and PCA based data aggregation algorithm. In addition, we will show that our method can reduce the amount of data transmission while ensuring a certain accuracy.

We randomly deploy a WSN with $N = 80$ sensors in a $100 * 100$m region to sample illumination intensity. The communication range of the sensor is a circular area within a radius of 20 meters. The value of the illumination

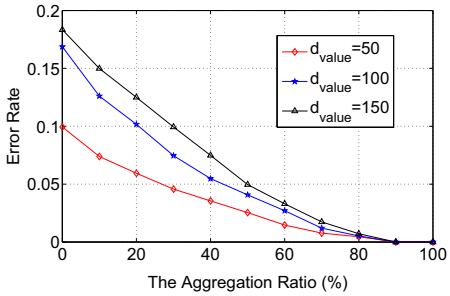

Fig. 3. Features of Principal Component Analysis.

Algorithm 2 Data Aggregation Algorithm

Input: Measurement data of clusters
Output: Aggregated data matrix;

1: Cluster head puts measurement data of its members into observation matrix x;
2: Calculate the covariance matrix $C = E[(x-E[x])(x-E[x])^T]$ of x;
3: Calculate the eigenvalues and the corresponding eigenvectors of matrix C;
4: Rank the eigenvalues and get the largest one;
5: Select m eigenvectors corresponding to the largest eigenvalue to form the transformation matrix P;
6: Compute the projection matrix $y = Px$;
7: Send the projection matrix y to the sink node.

intensity is produced by superposition of γ Gaussian distributions [22]

$$x(i,j,t) = 10^3 \sum_{k=1}^{\gamma} (g_{\mu_k,\Sigma_k}(i,j) e^{-\frac{t}{\delta_k}} \cos(2\pi f_k t)), \quad (17)$$

where (i,j) denotes a position of the sensing area. Then, $x(i,j,t)$ indicates the current illumination intensity of position (i,j) at time t.

In our simulation, we assume $\gamma = 2$, $f_1 = 0.1$Hz, $f_2 = 0.2$Hz, data means $\mu_1 = [3,2]$, $\mu_2 = [3,2]$, covariance matrixes $\Sigma_1 = [10,2;2,9]$, $\Sigma_2 = [8,2;2,10]$, and $\delta_1 = 10, \delta_2 = 15$. Such data can be a good summary of the data distribution of a variety of situations. Fig. 4 depicts the current value of data in each node. Other parameter settings are listed in Table I.

A. Evaluation of proposed clustering algorithm

According to the parameter settings in Table I, we can expect the optimal value $3 < k_{opt} < 17$ for 80-node network, where we set $c = 50$, $\varepsilon = 0.8$. Thus we vary the number of cluster members between 5 and 14.

Fig. 5 shows the average energy consumption per round with more than 95 percent accuracy. It is not difficult to observe that the experimental results and theoretical analysis are very similar, i.e., the optimal number of clusters by experiment is about 10-15 for 80-node network and the optimal number of clusters by theoretical analysis is 11.

TABLE I
VARIABLE PARAMETERS

Parameter	Value	Parameter	Value
E_{elec}	50nJ/bit	N	80
E_{amp}	1pJ/bit/m^4	M	100
d_{toSINK}	10~80m	E_{fs}	100pJ/bit/m^2

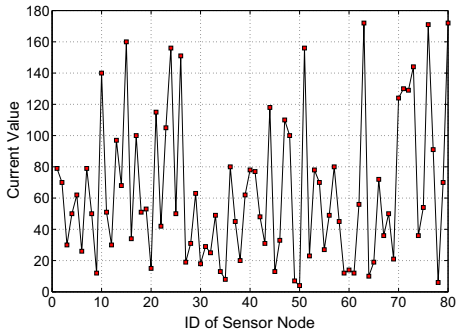

Fig. 4. Current data of nodes.

When there is only one cluster in the network, non-cluster head node will waste a lot of energy on data transmission. When there are too many clusters in our network, there are not much local data for data compression, which will influence the performance of compression algorithm.

Next, we employ the similarity criteria given in Section III-A to compute the similarity among nodes. Fig. 6 depicts the network topology obtained by similarity among sensors, in which the line connection between two nodes indicates that their data are similar. We can observe that nodes can effectively connect with their neighbor nodes with high similarity measurement, which means that our proposed similarity criteria can effectively eliminate data dissimilarity in a cluster. On the basis of the constructed cluster, our data aggregation algorithm based on PCA on cluster head nodes can deal with the similar data well.

Fig. 7 illustrates the results after clustering. It is clear that sensor nodes with higher similarity can be organized effectively into a cluster by our method, and the scope of data has also been well controlled in each cluster and the data are also relatively close between the node in one cluster. In this case, our clustering algorithm can effectively enhance the performance of the PCA based data aggregation algorithm to reduce the amount of transmitted data and the difficulty of data processing. The amount of aggregated data will be greatly less than the amount of the raw data generated by the sensor.

Another observation is that we can change the threshold c, ε and the normalized reconstruction error according to actual requirements. When we relax the thresholds, there are more opportunities for connecting more neighbor nodes. This helps the network effectively reduce computational cost since it leads to fewer clusters in the network.

B. Comparison of data aggregation algorithms

In this subsection, we compare the energy consumption of LEACH algorithm [16], K-means with principal component analysis (PCA) algorithm and our PCA based data aggregation algorithm. The reason that we choose LEACH algorithm is that our clustering algorithm is affected by LEACH algorithm. K-means with PCA algorithm is to compute K-means of observation data in the construction of transformation matrix P. In our simulation, each node begins with 2 J of energy, and communication capability of node is adjustable. Fig. 8 shows the relationship between energy consumption and the number of nodes. When the number of nodes is less than 300, the energy consumption of our algorithms is similar to LEACH and K-means PCA algorithm. When the number of nodes slowly increases, the amount of data increase as well. Our proposed algorithm can reduce significantly The energy consumption compared with other two methods. This is because (i) our similarity clustering approach has laid a good foundation for the data compression; (ii) by taking advantage of the similarity of data in network clustering, we can effectively reduce the network energy consumption for the given data accuracy.

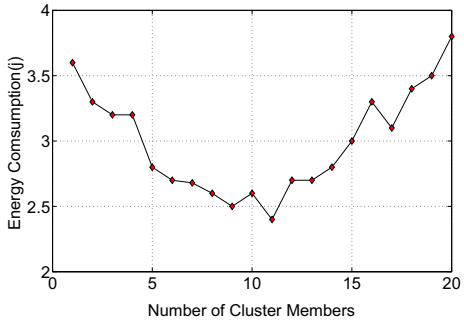

Fig. 5. The average energy consumption with $N = 80$ sensors in a $100m * 100m$ region

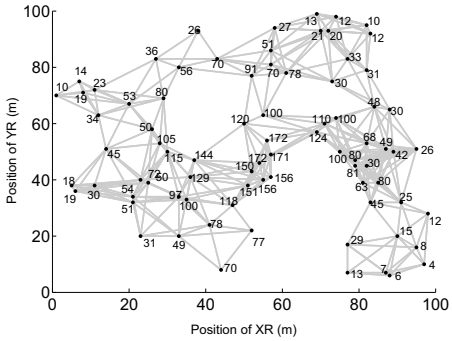

Fig. 6. Network topology with 80 sensors distributed in a 100*100 area.

VI. CONCLUSIONS

In this paper, based on the similarity of data among adjacent nodes, we propose a distributed clustering algorithm which can effectively organize the nodes with high similarity into a cluster for data aggregation, while ensuring uniform energy consumption within the cluster. Moreover, we propose the data aggregation algorithm based on principal component analysis (PCA) which can

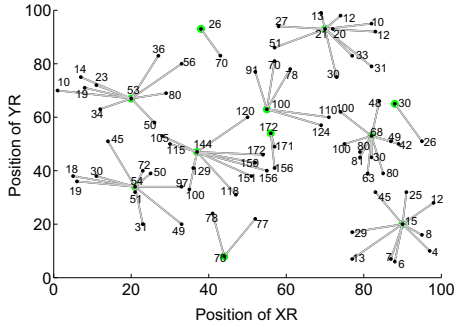

Fig. 7. Cluster structure by proposed clustering algorithm with $d = 50$.

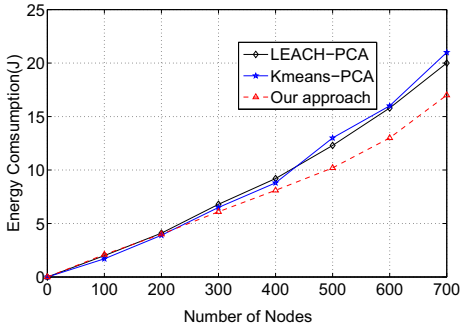

Fig. 8. Energy Consumption.

be executed in the cluster head. In particular, the proposed data aggregation algorithm will be more effective when it is combined with the clustering algorithm. Finally, our experimental results illustrate that the proposed algorithm can effectively reduce the amount of data transmission and energy consumption in the network.

VII. Acknowledgements*

This work was supported by the Fundamental Research Funds for the Central Universities (XDJK2013C094, XDJK2013A018, and XDJK2016A011), Natural Science Key Foundation of Chongqing (cstc2015jcyjBX0094), and the National Natural Science Foundation of China (no. 61373179, 61373178, and 61402381).

References

[1] J. Xu, S. Guo, B. Xiao, and J. He, "Energy-efficient big data storage and retrieval for wireless sensor networks with nonuniform node distribution," *Concurrency & Computation Practice & Experience*, vol. 27, no. 18, pp. 5765–5779, 2015.

[2] H. C. Weng, Y. H. Chen, E. H. K. Wu, and G. H. Chen, "Correlated data gathering with double trees in wireless sensor networks," *IEEE Sensors Journal*, vol. 12, no. 5, pp. 1147–1156, May 2012.

[3] W. Li, M. Bandai, and T. Watanabe, "Tradeoffs among delay, energy and accuracy of partial data aggregation in wireless sensor networks," in *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*, April 2010, pp. 917–924.

[4] B. Li, F. Gao, X. Liu, and X. Wang, "Improved distributed compressed sensing for smooth signals in wireless sensor networks," in *2016 International Conference on Computer, Information and Telecommunication Systems (CITS)*, July 2016, pp. 1–5.

[5] C. Guo and J. D. B. Nelson, "Compressive imaging with complex wavelet transform and turbo amp reconstruction," in *Signal Processing Conference (EUSIPCO), 2015 23rd European*, Aug 2015, pp. 1751–1755.

[6] C. Zhao, W. Zhang, X. Yang, Y. Yang, and Y. Q. Song, "A novel compressive sensing based data aggregation scheme for wireless sensor networks," in *2014 IEEE International Conference on Communications (ICC)*, June 2014, pp. 18–23.

[7] J. Fang, H. Li, Z. Chen, and Y. Gong, "Joint precoder design for distributed transmission of correlated sources in sensor networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2918–2929, June 2013.

[8] P. Wang, R. Dai, and I. F. Akyildiz, "A spatial correlation-based image compression framework for wireless multimedia sensor networks," *IEEE Transactions on Multimedia*, vol. 13, no. 2, pp. 388–401, April 2011.

[9] V. W. Mahyastuti and A. A. Pramudita, "Energy consumption evaluation of low energy adaptive clustering hierarchy routing protocol for wireless sensor network," in *Communication, Networks and Satellite (COMNETSAT), 2013 IEEE International Conference on*, Dec 2013, pp. 6–9.

[10] C. Wang, S. Tang, X. Y. Li, and C. Jiang, "Selectcast: Scalable data aggregation scheme in wireless sensor networks," in *INFOCOM, 2011 Proceedings IEEE*, April 2011, pp. 296–300.

[11] C. Hua and T. S. P. Yum, "Optimal routing and data aggregation for maximizing lifetime of wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 4, pp. 892–903, Aug 2008.

[12] R. Barton and R. Zheng, "Order-optimal data aggregation in regular wireless sensor networks," *Information Theory, IEEE Transactions on*, vol. 56, no. 11, pp. 5811–5821, Nov 2010.

[13] C. Liu, K. Wu, and J. Pei, "An energy-efficient data collection framework for wireless sensor networks by exploiting spatiotemporal correlation," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 18, no. 7, pp. 1010–1023, July 2007.

[14] Z. Chen, S. Yang, L. Li, and Z. Xie, "A clustering approximation mechanism based on data spatial correlation in wireless sensor networks," in *Wireless Telecommunications Symposium (WTS), 2010*, April 2010, pp. 1–7.

[15] L. G. Rios and J. A. I. Diguez, "Big data infrastructure for analyzing data generated by wireless sensor networks," in *2014 IEEE International Congress on Big Data*, June 2014, pp. 816–823.

[16] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *Wireless Communications, IEEE Transactions on*, vol. 1, no. 4, pp. 660–670, Oct 2002.

[17] L. Du, B. Wang, P. Wang, Y. Ma, and H. Liu, "Noise reduction method based on principal component analysis with beta process for micro-doppler radar signatures," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 8, pp. 4028–4040, Aug 2015.

[18] N. Ghadban, P. Honeine, C. Francis, F. Mourad-Chehade, and J. Farah, "Strategies for principal component analysis in wireless sensor networks," in *Sensor Array and Multichannel Signal Processing Workshop (SAM), 2014 IEEE 8th*, June 2014, pp. 233–236.

[19] A. Rooshenas, H. Rabiee, A. Movaghar, and M. Naderi, "Reducing the data transmission in wireless sensor networks using the principal component analysis," in *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2010 Sixth International Conference on*, Dec 2010, pp. 133–138.

[20] F. Chen, F. Wen, and H. Jia, "Algorithm of data compression based on multiple principal component analysis over the wsn," in *Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on*, Sept 2010, pp. 1–4.

[21] K. M. Mackenthun, "Covariance matrix properties and multiple symbol/soft decision detection in slow flat rician fading," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 2, pp. 263–275, Feb 1995.

[22] N. Sun and Q. Lian, "Data aggregation technique combined temporal-spatial correlation with compressed sensing in wireless sensor networks," in *Conference Anthology, IEEE*, Jan 2013, pp. 1–5.