Data compression algorithms for sensor networks with periodic transmission schemes

Jianxin Chen[1], *Pengcheng* Wang[1], *Xinzhuo* Ren[1], *Haojie* Meng[1], *Yinfei* Xu[1], and *Jiaqiang* Zhuge[2,*]

[1]Hangzhou Electric Equipment Manufacturing Co., Ltd. Hangzhou, China
[2]School of Automation, Hangzhou Dianzi University, Hangzhou, China

Abstract. The operating state of switch cabinet is significant for the reliability of the whole power system, collecting and monitoring its data through the wireless sensor network is an effective method to avoid accidents. This paper proposes a data compression method based on periodic transmission model under the condition of limited energy consumption and memory space resources in the complex environment of switch cabinet sensor networks. Then, the proposed method is rigorously and intuitively shown by theoretical derivation and algorithm flow chart. Finally, numerical simulations are carried out and compared with the original data. The comparisons of compression ratio and error results indicate that the improved algorithm has a better effect on the periodic sensing data with interference and can make sure the change trend of data by making certain timing sequence.

Keywords: Wireless sensor network, Data compression, Periodic transmission model, Pearson correlation coefficient, Outlier.

1 Introduction

With the rapid development of power grid in China and the increase of maintenance workload in power system, switch cabinet, as one of the key equipment of power system, its operating state and maintenance have a vital influence on the whole power system. Collecting and monitoring data of the operating state of the switch cabinet through wireless sensor networks is an effective method to reduce human resources and avoid accidents [1]. However, the resources, such as energy consumption, storage space, communication bandwidth and processing speed, are always limited in wireless sensor networks [2,3]. Given that the energy consumption of sensor processing is much lower than that of sensor communication and the sensor information has a lot of data redundancy, thus compressing and then transmitting data is an effective way to save energy consumption of wireless sensors [4].

The main purpose of this paper is to solve the problem of limited sensor resources in wireless sensor networks of switch cabinets, and to save energy consumption and storage

* Corresponding author: 1348889002@qq.com

space of sensor nodes by data compression algorithms. Compared with the traditional data compression algorithms, the size of wireless sensor network data compression algorithm proposed in this paper is small and the complexity is low. The data compression algorithm based on the periodic transmission model has obvious advantages like more compression ways and higher compression rate than the traditional algorithms, so it has received extensive and intense attentions in recent years [5-8].

However, some data compression algorithms based on periodic transmission model cannot guarantee the time sequence of data [9], which results in missing of many sensor data features, such as the evolution trend of sensor data along with time. In view of this problem, this paper introduces Pearson correlation coefficient and proposes a data compression algorithm which can guarantee the time sequence, especially for smooth and periodic data. Furthermore, the concept of outlier is introduced to replace the incorrect or unnecessary data caused by interference, so as to achieve better data compression effect. Finally, compile the processed data into dictionary to reduce the number of data bits, and then complete the data compression.

2 Algorithm design

2.1 Periodic transmission model

In sensor networks, collecting and sending data to the next node continuously is not effective, which causes the large amount of sensor energy loss and communication resource waste. Therefore, a data transmission model based on periodic collection is proposed for sensor nodes, which allows sensor nodes to collect and clean up a series of data before sending to the next node. Because a lot of time data redundancy generated in the periodic data transmission model, data compression is an effective method to save sensor energy.

The following is a specific algorithm generated by periodic data transmission models.

The sensor node collects the data in the current period, when each reading is collected, then the number of readings is added "1", and when τ reading is obtained, the reading vector is constructed according to chronological order as follows:

$$R = [r_1, r_2, \ldots, r_\tau]$$

where τ is the total number of readings for the current period, $\tau = 2^N, N \in Z$.

2.2 Data compression algorithm based on Pearson correlation coefficients

2.2.1 Pearson correlation coefficients

Pearson correlation coefficients represent the relevance degree between the two data sets, and the range is between [-1,1]. The higher the Pearson correlation coefficient is, the higher the relevance degree is. When the Pearson coefficient equals to 1, which means that the two data sets are completely positive correlated, that is,$y = ax + b$, y represents the vector form of the first data set in order, x represents the vector form of the second data set in order, a is an arbitrary positive number, b is an arbitrary constant. When the Pearson coefficient equals to -1, the two data sets are completely negatively correlated. By contrast, when it equals to 0, the two data sets are completely independent.

The Pearson correlation coefficient is described as follows:

$$\rho_{R_i,R_j} = \frac{n\Sigma r_i r_j - \Sigma r_i \Sigma r_j}{\sqrt{n\Sigma r_i^2 - (\Sigma r_i)^2}\sqrt{n\Sigma r_j^2 - (\Sigma r_j)^2}} \quad (1)$$

where $r_i \in R_i, r_j \in R_j$, n is the number of elements of R_i or R_j.

In the case of complete positive correlation between two data sets, if their element average is equal, it can be obtained that the data set relation as $y = x$, which means the two data sets are completely equal.

The absolute value of the average difference of the elements is expressed as follows:

$$d_{R_i,R_j} = \left|\frac{\sum r_i - \sum r_j}{n}\right| \quad (2)$$

2.2.2 Data compression algorithm based on Pearson correlation coefficients

The key idea of the algorithm is to divide the vector R_i into two subvector $R_{i_1} = [r_{i_1}, r_{i_2}, \ldots, r_{i_n}]$ and $R_{i_2} = [r_{i_{n+1}}, r_{i_{n+2}}, \ldots, r_{i_{2n}}]$ with the same number of elements, and then to estimate whether to compress the vector or not by the comparisons between $\rho_{R_{i1},R_{i2}}$and $d_{R_{i1},R_{i2}}$as well as t_p and t_m. More details are given as follows:

When $2n \neq 2$: as $\rho_{R_{i1},R_{i2}} \geq t_p$ and $d_{R_{i1},R_{i2}} \leq t_m$, it is considered that these two vectors highly correlated and numerically similar. t_p represents the high correlation threshold values, $t_p \in [-1,1]$, $t_m \geq 0$ represents the threshold values closed to the average. The higher the value of t_p is, the more accurate the data is, while the lower the value of t_m is, the higher the compression ratio is.

The vector R_{i_3} whose elements are the average of the corresponding elements of two sub-vectors:

$$R_{i_3} = \left[\frac{r_{i_1}+r_{i_{n+1}}}{2}, \frac{r_{i_2}+r_{i_{n+2}}}{2}, \ldots, \frac{r_{i_n}+r_{i_{2n}}}{2}\right] \quad (3)$$

Then update R_i as $R_i = [R_{i_3}, R_{i_3}]$ and the value of the corresponding position R_i in the reading vector R;

When $\rho_{R_{i1},R_{i2}} < t_p$ or $d_{R_{i1},R_{i2}} > t_m$, it is considered that R_{i_1} is not highly positively correlated with R_{i_2} or the average value of vector elements is not close, then keep the original value.

In particular, for the case of $2n = 2$, Pearson correlation coefficients are not applicable, so the absolute values of the difference between two elements in the original vector R_i can be directly compared, that is, the magnitude relationship between $|r_{i_1} - r_{i_2}|$ and t_m.

To be specific, when $|r_{i_1} - r_{i_2}| \leq t_m$, the values of two elements in R_i are similar, the let

$$R_i = \left[\frac{r_{i_1}+r_{i_2}}{2}, \frac{r_{i_1}+r_{i_2}}{2}\right] \quad (4)$$

Likewise, update the value of the corresponding position R_i in the reading vector R. For the case with $|r_{i_1} - r_{i_2}| > t_m$, the original value is maintained.

Based on the above developments, construct the vector queue V_R to be executed, add the unexecuted vectors R_{i_1}, R_{i_2} and R_{i_3}, delete the executed original vectors R_i, and select the original vectors according to the order of addition to execute the above algorithm.

2.3 Improved data compression algorithm with outlier replacement

On the basis of the above algorithm, the concept of outlier is introduced. The outlier refers to the large difference between one or more data and other data in the data set, so it needs to be eliminated or replaced. Given that the number of vector elements does not match in the proposed algorithm if we remove the outlier directly, thus we adopt the method of replacing outlier. Meanwhile, it is also important to estimate whether the value is outlier or not, thus some data are chosen as alternate outliers in advance, then judge them as outliers by testing, and finally replace the associated data. The details are provided as follows.

In particular, for the case of $2n = 4$, if $\rho_{R_{i_1},R_{i2}} < t_p$ or $d_{R_{i_1},R_{i2}} > t_m$, and the absolute value of the difference between two elements in R_{i_1} and R_{i_2} is greater than t_m, the reading element $\mathrm{r_i}$ in the sub-vectors R_{i_1} and R_{i_2} is regarded as the alternate outlier, and let R_{i_4}=$[R_{i_1}, R_{i_1}]$, R_{i_5}=$[R_{i_2}, R_{i_2}]$.

Then, calculate $\rho_{R_{i_4},R_i}$, $\rho_{R_{i_5},R_i}$, which are the Pearson correlation coefficients between R_{i_4} and vector R_i', R_{i_5} and vector R_i' respectively. Specifically, R_i' can be obtained as follows: take the remainder that obtained through the first element position of vector R_i in the reading vector set R divided by 8. If the remainder is 1, take 4 elements backward from the position after the tail element to form vector R_i'; if the remainder is 5, take 4 elements forward from the position before the first element to form vector R_i'.

In the following, it is required to judge the magnitude relationship between $\rho_{R_{i_4},R_i}$, $\rho_{R_{i_5},R_i}$, and t_m as well as the magnitude relationship between d_{R_{i4},R_i}, d_{R_{i5},R_i} and t_m respectively.

When $\rho_{R_{i_4},R_i} \geq t_p$ and $d_{R_{i4},R_i} \leq t_m$, it is considered that $\mathrm{R_{i_4}}$ and $\mathrm{R_i}'$ are highly correlated, the average value of elements is similar and the reading element $\mathrm{r_i}$ is an outlier. Similarly, when $\rho_{R_{i_5},R_i} \geq t_p$ and $d_{R_{i5},R_i} \leq t_m$, it indicates that R_{i_5} and R_i' are highly correlated, the average value of elements is closed, and the reading element r_i is an outlier. Note that when the calculated values of R_{i_4} and R_{i_5} both meet the above conditions, then take the value that R_{i_4} satisfied.

Following the above analysis, calculate the corresponding vector R_{i_3}, then update $[R_i, R_i'] = [R_{i_3}, R_{i_3}]$ and the corresponding values of R_i and R_i'in the reading vector R;

When $\rho_{R_{i_4},R_i} < t_p$ or $d_{R_{i4},R_i} > t_m$ and $\rho_{R_{i_5},R_i} < t_p$ or $d_{R_{i5},R_i} > t_m$ are satisfied, the reading element r_i is not considered as an outlier and the original value is maintained.

2.4 Compilation dictionary algorithms

On the basis of the above algorithm, the concept of outlier is introduced. The outlier refers to the large difference between one or more data and other data in the data set, so it needs to be eliminated or replaced. Given that the number of vector elements does not match in the proposed algorithm if we remove the outlier directly, thus we adopt the method of replacing outlier. Meanwhile, it is also important to estimate whether the value is outlier or not, thus some data are chosen as alternate outliers in advance, then judge them as outliers by testing, and finally replace the associated data. The details are provided as follows.

The above algorithm produces a lot of time data redundancy, but does not carry on the substantive data compression, so the compilation dictionary algorithm is a feasible data compression algorithm.

When the vector queue to be executed is empty, the data processing ends. Count the number of the same and different elements in the reading vector, respectively, and sort the number of the same elements from largest to smallest, then assign the binary index

according to the number of different elements, compile into the following dictionary, and the element reading in the reading vector R is replaced by the binary index as follows:

Table 1. Binary index.

Number of different elements $\boldsymbol{n_i}$	Index binary representation $\boldsymbol{s_1}$	Corresponding element readings $\boldsymbol{r_i}$
1, 2	0, 1	r_1, r_2
3, 4	00, 01, 10, 11	r_1, r_2, r_3, r_4
5, 6, 7, 8	000, 001, ..., 111	$r_1, r_2, ..., r_8$
...	...	...

Finally, transmit the resulting dictionary and reading vector R to the next sensor node. Enter the next period and circulate all the above steps. The overall algorithm structure diagram is shown in figure 1.

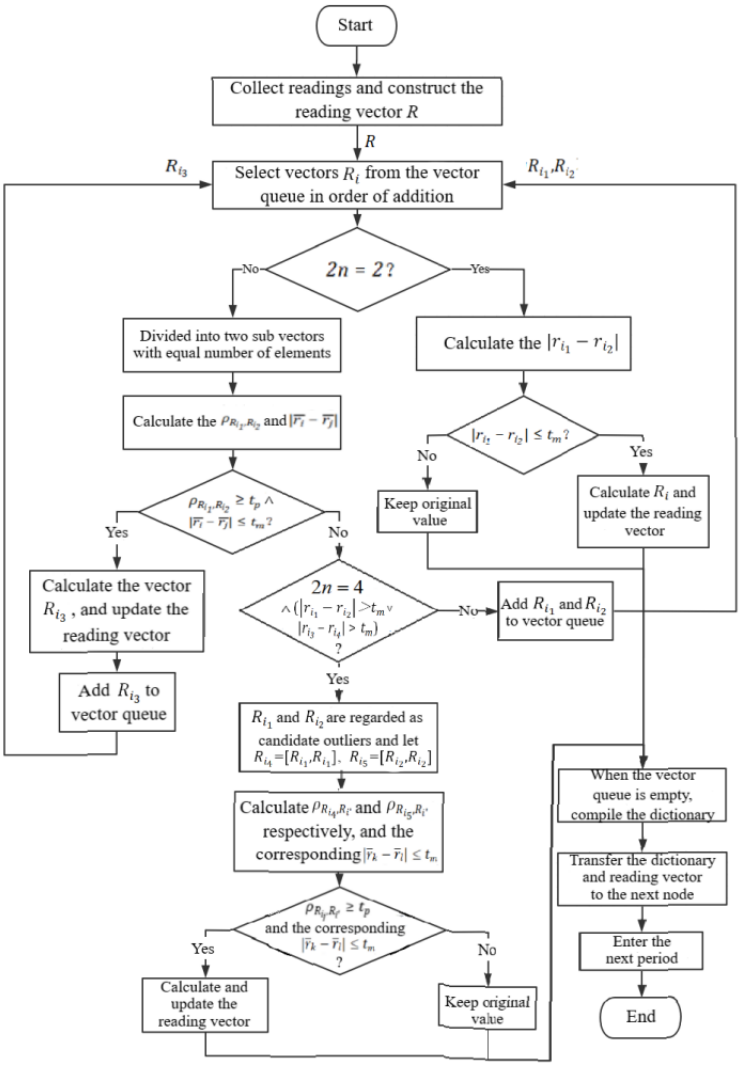

Fig. 1. General algorithm structure chart.

3 Simulations

Randomly generate a section of simulated data set with disturbance according to the above algorithm, and retain the undisturbed data set, and use the data compression algorithm to generate the compressed data set. Let $N = 8$, then one can gets $\tau = 256$. Sample 10 periods to generate trigonometric function simulation data, which is randomly superposed of normal distribution perturbations with $N(1,0.04)$, and then let $t_p = min\{(N_i - 2) * 0.35 - 1,1\}, t_m = 0.6$, where $N_i = log_2(2n)$ represents the number of elements of the currently executed vector R_i.

The simulation results show that the data compression is 35.90% of the original data, the average absolute difference between the compressed and the uninterfered data is 0.1374, and the one between the compressed and the collected data is 0.1352, and between the collected data and the uninterfered data is 0.1580. Note that the average absolute difference value approximates to that between compressed data related and disturbance, this indicates that the algorithm can keep good compression data rate while maintaining low distortion rate. Hence, the proposed algorithm is effective for wireless sensor network (WSN) for switch cabinet, and can greatly save the energy consumption, storge space and other problems of the limited sensor resources.

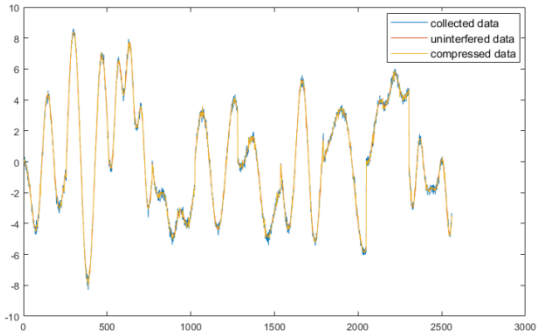

Fig. 2. Three types of data sets.

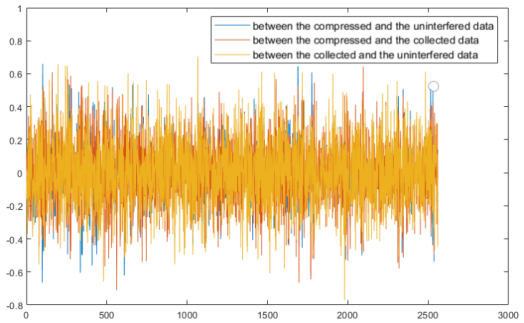

Fig. 3. Three types of data deviation.

4 Conclusion

The data compression algorithm proposed in this paper can achieve good effect in processing the sensing data of the periodic switch cabinet with disturbances, which can compress the network data to 20%-40% of the original data, and guarantee the lower distortion rate. The more the total number of readings in per reading period, the higher the compression rate. Since the method proposed in the paper guarantees the timing sequence,

the sensor data of the switch cabinet also guarantees the evolution trend of time. At the same time, adjust the two thresholds t_p and t_m reasonably according to the specific situations, so that the compression method can achieve different effects flexibly.

References

1. Y. Xu, L. Chen, L. Gan and Y. Zhang. Principle and Application of Wireless Sensor Network Technology, Tsinghua University Press, 2019.
2. G. Anastasi, M. Conti, M. Di Francesco, A. Passarella. Energy conservation in wireless sensor networks: A survey, Ad Hoc Networks, vol. 7, no. 3, pp. 537-568, 2009.
3. S. Bhandari, N. Bergmann, R. Jurdak, and B. Kusy. Time series data analysis of wireless sensor network measurements of temperature. Sensors, vol. 17, no. 6, p. 1221, May 2017.
4. N. Kimura, and S. Latifi. A survey on data compression in wireless sensor networks. Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC05), 2005.
5. P. Prasad. Recent trend in wireless sensor network and its applications: A survey. Sensor Review, vol. 35, no 2, pp. 229-236, 2015.
6. C. B. Mudgule, U. Nagaraj, and P. Ganjewar. Data compression in wireless sensor network: A survey. International Journal of Innovative Research in Computer and Communication Engineering, vol. 2, 6664-6673, 2014.
7. J. Wang, Y. Gao, W. Liu, W. Wu, and S.-J. Lim. An asynchronous clustering and mobile data gathering schema based on timer mechanism in wireless sensor networks. Comput., Mater. Continua, vol. 58, no. 3, pp. 711–725, 2019.
8. M. Y. Durrani, R. Tariq, F. Aadil, M. Maqsood, Y. Nam, and K. Muhammad. Adaptive node clustering technique for smart ocean under water sensor network (SOSNET). Sensors, vol. 19, no. 5, p. 1145, Mar. 2019.
9. H. Harb, C. A. Jaoude. Combining Compression and Clustering Techniques to Handle Big Data Collected in Sensor Networks. IEEE Middle East and North Africa Communications Conference, 2018.