Research on energy-saving algorithm of wireless sensor network based on penalty error matrix

ZHANG Ping¹, LIU Meiqing¹, SU Kai¹, HAN Shuhuan¹, CHEN Wei²

¹College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, China
²National Demonstration Center for Experimental Electrical and Control Engineering Education, Lanzhou 730050, China

Abstract: Taking into account this problems of large data volume, high frequency, and fast energy consumption in data transmission and reception of wireless sensor networks, an energy-saving algorithm based on a penalty error matrix is adopted to reduce network energy consumption. By introducing a penalty error matrix in the adaptive cubic exponential smoothing algorithm, using the principle of the minimum sum of squares of error to track the time series in real time, adjusting and obtaining the optimal smoothing coefficient in time, and it has strong adaptability to data with a large fluctuation range. The MATLAB simulation experiment results show that when the time slot range is 1500, compared with the traditional cubic exponential smoothing algorithm and the adaptive cubic exponential smoothing algorithm, the energy saving effect and prediction accuracy of this algorithm are significantly improved.

1. Introduction
Zigbee Wireless Sensor Network (Wireless Sensor Networks, WSN) [1-3] is generally used in smart homes [4], water quality monitoring [5], health services [6-7], industrial monitoring [8-9], environmental monitoring And other fields. Zigbee usually uses battery power supply, and the energy consumption of nodes in the network is relatively large. In many occasions, due to the influence of geographical location, environment and other factors, it is extremely difficult to replace the battery, which makes the system operation face greater problems. Therefore, reducing the energy loss in wireless sensor networks will be an important problem that needs to be solved urgently [10], and it is also a very challenging problem.

Each node of the zigbee wireless sensor network exchanges information through a multi-hop network environment. In the current research, there are few methods to reduce the power consumption in the multi-hop wireless sensor network environment. Therefore, reducing the energy consumption of the sensing node has become a key to extending battery life. Literature [11] proposes an algorithm that combines MAC monitoring to perform periodic sleep. However, the algorithm does not consider the problems caused by different node logic types. Literature [12] proposed an asynchronous sleep algorithm to detect the amount of remaining power, but it ignored the coordination of data transmission and sleep in actual situations. Literature [13] proposes a way to put redundant nodes to sleep, and then adopt a phased wake-up method to reduce data collection, thereby saving energy. This strategy can improve network performance and extend network life, but it may produce perception blind spots and cause waste of resources. Literature [14] proposed a dynamic strategy to reduce node energy consumption, through the study of local time synchronization, to prevent local network
communication conflicts. This method ignores the influence of other collected objects and lacks comprehensive persuasiveness.

During the normal operation of the wireless sensor network, each node must frequently turn on/off the transceiver to receive or send signals. This process will cause unnecessary energy waste. Aiming at this problem, taking the zigbee multi-hop network as the research object, an improved adaptive cubic exponential smoothing algorithm[15] is used to reduce the energy loss in the network. Firstly, historical data is collected, and the data of the receiving node and the sending node are simultaneously predicted. Then the time series are tracked in real time through the principle of the minimum sum of squares of errors, and the penalty error matrix is introduced for refinement processing to obtain the optimal smoothing coefficient. Finally, the simulation experiment verifies that the method has better energy-saving effect and higher prediction accuracy.

2. Traditional cubic exponential smoothing algorithm prediction model

The triple exponential smoothing algorithm is to perform another smoothing based on the second smoothing. Its basic smoothing model can be divided into the following two types:

1) Smoothing of historical data

The smoothing formula for historical data is:

\[
\begin{align*}
S_{t}^{(1)} &= \beta X_{t} + (1 - \beta)S_{t-1}^{(1)} \\
S_{t}^{(2)} &= \beta S_{t}^{(1)} + (1 - \beta)S_{t-1}^{(2)} \\
S_{t}^{(3)} &= \beta S_{t}^{(2)} + (1 - \beta)S_{t-1}^{(3)}
\end{align*}
\]  

In this formula: \( t = 2, 3 \), \( S_{t}^{(1)} \), \( S_{t}^{(2)} \), \( S_{t}^{(3)} \) respectively represent the first, second, and third exponential smoothing values, \( \beta \) \((0 < \beta < 1)\) is the static smoothing factor, and \( \alpha \in (0, 1) \). \( X \) is the actual data of the \( t \) period of the time series \( t \).

2) Smoothing of future data

The smoothing formula for future data is:

\[
\begin{align*}
a_{i} &= 3S_{t}^{(1)} - 3S_{t}^{(2)} + S_{t}^{(3)} \\
b_{i} &= \frac{\beta}{2(1 - \beta)^2} ((6 - 5\beta)S_{t}^{(1)} - (10 - 8\beta)S_{t}^{(2)} + (4 - 3\beta)S_{t}^{(3)}) \\
c_{i} &= \frac{\beta^2}{2(1 - \beta)^2} (S_{t}^{(1)} - 2S_{t}^{(2)} + S_{t}^{(3)})
\end{align*}
\]  

In this formula: \( a_{i}, b_{i}, c_{i} \) are 3 prediction coefficients. Let \( T \) is the number of time points predicted forward from time \( t \).

The forecast for the period is:

\[ Y_{t+T} = a_{i} + b_{i}T + c_{i}T^2 \]  

The prediction accuracy of the exponential smoothing prediction model is closely related to the value of the smoothing coefficient, so the selection of the smoothing coefficient is the key to establishing a smoothing exponential model. Choose a suitable smoothing coefficient so that the smaller the error between the predicted data and the measured data, the better the prediction effect. Based on this principle, choose to calculate the error sum of squares, and the minimum formula of the error sum of squares is:

\[ f = \min_{\alpha} \sum_{t=4}^{N} (Y_{t} - X_{t})^2 \]  

In formula (4), \( N \) is the number of historical data periods in each forecast, and \( Y_{t} \) is the \( t \)-th period forecast data of the time series \( \{X_{t}\} \). \( \{X_{t}\} \) is the actual data of the \( t \)-th period of the time series \( t \).
When using the cubic exponential smoothing model, you need to determine the initial value of the smoothing model and the smoothing coefficient. The expression for determining the initial value is:

$$S_0^{(1)} = S_0^{(2)} = S_0^{(3)} = (X_t + X_{t+1} + X_{t+2}) / 3$$  \hspace{1cm} (5)$$

In this formula: \((X_t + X_{t+1} + X_{t+2}) / 3\) represents the average of the first 3 periods of historical data.

In the calculation process, the initial value of \(\beta\) can be set to 0.0001 to obtain the sum of squared errors of all known historical data \(X_t\) and the predicted value \(Y_t\) of the corresponding period. Set the step size \(\lambda = 0.001\) and recalculate the sum of squared errors when \(\beta = 0.0011\). In such a cycle, the optimal \(\beta\) can be obtained according to equation (4), which can be used \(\beta\) to predict future data.

3. Adaptive cubic exponential smoothing algorithm

The smoothing coefficient \(\beta\) reflects the proportion of historical data in the exponential smoothing value in different periods, that is, the exponential smoothing method will assign different weights to the historical observation data according to the influence of the past observation data. However, the size of the smoothing coefficient is related to the distance from the predicted value: the farther from the predicted value, the larger the smoothing coefficient \(\beta\). The parameters in the three-time exponential smoothing algorithm are fixed values, so that the data cannot be changed in time, thus affecting the accuracy of prediction. The adaptive cubic exponential smoothing algorithm will be improved according to the optimal selectivity of the smoothing coefficient, using 0.001 as the step size factor to filter out the smoothing coefficient that minimizes the error. Expand the formula (1) to get:

$$S_t^{(1)} = \sum_{i=1}^{t} \beta(1-\beta)^{i-1} X_t + (1-\beta)S_0^{(1)}$$

$$S_t^{(2)} = \sum_{i=1}^{t} \beta(1-\beta)^{i-1} S_t^{(1)} + (1-\beta)S_0^{(2)}$$

$$S_t^{(3)} = \sum_{i=1}^{t} \beta(1-\beta)^{i-1} S_t^{(2)} + (1-\beta)S_0^{(3)}$$  \hspace{1cm} (6)$$

In formula: \(\sum_{i=1}^{t} (1-\beta)^{i-1} = 1 - (1-\beta)^t \neq 1\). Let \(k\) be the number of predictions, \(\beta_k\) is the adaptive smoothing coefficient of the \(K\)th prediction. Normalize the smoothing coefficients in the traditional model and expand them appropriately to get

$$\varphi_{k,t} = \frac{\beta_k}{1-(1-\beta_k)^t}$$  \hspace{1cm} (7)$$

In formula: \(K\) is the number of predictions, and \(\beta_k\) is the \(K\)-th adaptive smoothing coefficient, and \(\beta_k \in (0,1)\). When \(t \geq 1\), \(\varphi_{k,t} \in (0,1)\), satisfies the smoothing factor condition. In formula (7), \(\varphi_{k,t}\) is a function of \(t\). Therefore, \(\varphi_{k,t}\) can be considered as the dynamic smoothing coefficient for the \(K\)th prediction.

1) Smoothing of historical data

The smoothing formula of historical data can be rewritten as:

$$S_t^{(1)} = \varphi_{k,t} X_t + (1-\varphi_{k,t})S_{t-1}^{(1)}$$

$$S_t^{(2)} = \varphi_{k,t} S_t^{(1)} + (1-\varphi_{k,t})S_{t-1}^{(2)}$$

$$S_t^{(3)} = \varphi_{k,t} S_t^{(2)} + (1-\varphi_{k,t})S_{t-1}^{(3)}$$  \hspace{1cm} (8)$$

Let \(X_t = X_{t-1}, \ t = 1, 2 \cdots N, \ N\) is the number of historical data periods selected for each forecast. Minimum error sum of squares:
Through formula (9), an adaptive smoothing coefficient $\beta_k$ can be determined every time, and then continue to use the principle of minimum sum of squares for subsequent prediction.

2) Smoothing of future data

Change formula (2) to

$$
\begin{align*}
&\begin{bmatrix}
    a_{k,t} = 3S_{k,t}^{(1)} - 3S_{k,t}^{(2)} + S_{k,t}^{(3)} \\
b_{k,t} = \frac{\varphi_{k,t}}{2(1-\varphi_{k,t})} \left((6 - 5\varphi_{k,t})S_{k,t}^{(1)} - (10 - 8\varphi_{k,t})S_{k,t}^{(2)} + (4 - 3\varphi_{k,t})S_{k,t}^{(3)}\right) \\
c_{k,t} = \frac{\varphi_{k,t}^2}{2(1-\varphi_{k,t})} (S_{k,t}^{(1)} - 2S_{k,t}^{(2)} + S_{k,t}^{(3)})
\end{bmatrix}
\end{align*}
$$

According to formula (9), the smoothing coefficient $\beta_k$ of the previous N periods is determined, and then the data of the $k+N-1+T$ period is predicted by the adaptive cubic exponential smoothing algorithm. The predicted value at time $t+T$ is:

$$\hat{Y}_{k,t+T} = a_{k,t} + b_{k,t}T + c_{k,t}T^2$$

4. Prediction algorithm based on penalty error matrix

4.1. Penalty error matrix

In order to represent the penalty from the current states to the next state in real time, this paper introduces a penalty error matrix $Q$. When the error sum of squares is the smallest, the state is set to A, and the corresponding sub-item in the matrix Q is +2; otherwise, the state is set to B, and the reward value of Q is -1. The matrix update formula is as follows:

$$Q(s,a) = R(s,a) + \gamma \cdot Q(s,all\ actions)$$

In the formula: $\gamma$ represents the influence weight of the error matrix on the result. Here, let $\gamma = 0.2$, and $a$ is the element of Q. Besides, all actions represent all states.

4.2. Improvement of adaptive cubic exponential smoothing algorithm

Due to the limited computing power of the wireless sensor network, the number of data groups sent from the sensing node to the routing node is limited to 5 groups. If the sum of squared errors is the smallest, then the $Q_i$ (the ith row and the first column of the penalty error matrix Q) gets +2 points; otherwise, $Q_i$ gets -1 points. The number of iterations is set to 10, that is, when the number of times the zigbee sensing node sends to the routing node is 10, a new penalty error matrix Q is formed through iterative calculation, and its form is:

$$Q = \begin{bmatrix}
0 & 0 \\
0 & 0 \\
\vdots & \vdots \\
0 & 0 \\
\end{bmatrix} \xrightarrow{\gamma} \begin{bmatrix}
a_{11} & a_{12} \\
a_{21} & a_{22} \\
\vdots & \vdots \\
a_{51} & a_{52}
\end{bmatrix}$$

Where: $a_i (i = 1,2,...,5; j = 1,2)$ is the reward value of each iteration. If the sum of all penalties is less than or equal to 0, parameter optimization is not performed, and vice versa. Find the variance of Q in formula (10):
\[ R = \sum_{i=1}^{n} (\hat{a}_i - a_i)^2 \]  

(14)

In the formula, \( a_i \) is the element in \( Q \), and the variance of all elements between the current matrix \( Q \) and the previous matrix \( \hat{Q} \) is summed. Design a threshold \( \hat{L} \), if \( R < \hat{L} \), add an adjustment factor \((1 - R/R) \beta \); conversely, if \( R \) is greater than the designed threshold \( \hat{H} \), reduce an adjustment factor to \( \beta \), if \( \hat{L} < R < \hat{H} \), no adjustment is required. Among them \( L = 0.05R, H = 0.5R \), \( \hat{L}, \hat{R}, \hat{H} \) represents the variance obtained from the matrix \( \hat{Q} \) of the previous iteration, and \( L, R, H \) represents the variance of the matrix \( Q \).

5. Experiment and analysis

Energy consumption gain:

\[ \eta = \frac{E_{DA}}{E} \]  

(15)

In the formula: \( E_{DA} \) represents the energy consumption saved by the communication algorithm, and \( E \) represents the energy consumption of the dry battery. The experimental conditions are shown in Table 1.

| Perception node (a) | Routing node (a) | Coordination node (a) | Data forwarding rate (Mb/s) | Channel bandwidth (Mb/s) | Initial energy mJ | Time (time slot) |
|---------------------|------------------|-----------------------|-----------------------------|--------------------------|-------------------|-----------------|
| 6                   | 1                | 1                     | 1                           | 3                        | 1100              | 1500            |

According to the remaining energy, this paper compares the traditional cubic exponential smoothing algorithm with the adaptive cubic exponential smoothing algorithm and the adaptive cubic exponential smoothing algorithm based on the penalty error matrix. The number of nodes is different, the energy consumption will be different, the different numbers of sensing nodes are simulated and analyzed, and the comparison is shown in Figures 1 and 2.

It can be seen from Figure 2 that the adaptive cubic exponential smoothing algorithm saves about 5% of energy consumption compared with the traditional cubic exponential smoothing algorithm, and the energy saving effect is not good. Compared with the algorithm after introducing the penalty error matrix, the adaptive cubic exponential smoothing algorithm has a larger amount of calculation at the beginning, so the energy consumption drops faster. However, with the increase of time slots, the speed of decline becomes slower compared with the original algorithm, energy consumption is saved by about 13%, and the remaining energy is greatly improved compared with the previous two methods. In Figure 3, first obtain the energy consumption ratio by formula (12), and then analyze the curve. It can be seen that when the number of sensing nodes is less than or equal to 12, the number of sensing nodes...
is proportional to the energy saving. When the number of sensing nodes continues to increase, the energy saving ratio of the algorithm after introducing the penalty error matrix also increases. When the number of nodes is greater than 12, the curve tends to be stable, and the energy saving ratio remains basically unchanged.

This paper makes predictions for 600 sensing nodes, and the relative error comparison table of the prediction results is shown in Table 2:

| Number of nodes | Adaptive cubic exponential smoothing algorithm | Algorithm after introducing penalty error matrix |
|-----------------|-----------------------------------------------|-----------------------------------------------|
| 50              | 0.1506                                        | 0.1343                                        |
| 100             | -0.1511                                       | 0.1189                                        |
| 150             | 0.1469                                        | 0.1080                                        |
| 200             | -0.1468                                       | -0.0873                                       |
| 250             | 0.1466                                        | 0.0861                                        |
| 300             | 0.1443                                        | -0.0728                                       |
| 350             | 0.1462                                        | 0.0755                                        |
| 400             | -0.1438                                       | -0.0743                                       |
| 450             | 0.1431                                        | 0.0752                                        |
| 500             | -0.1412                                       | 0.0736                                        |
| 550             | -0.1422                                       | 0.0733                                        |
| 600             | 0.1423                                        | 0.0738                                        |
| Average relative error/% | 0.1454                                       | 0.0877                                        |

Among them: the relative error of the prediction result = (Predicted value-actual measured value)/actual measured value×100%; the average relative error is the absolute value of the relative error and then the average value.

It can be seen from the table that the adaptive cubic exponential smoothing algorithm based on the penalty error matrix is compared with the adaptive cubic exponential smoothing algorithm. At the beginning, there is not much difference between the two. As the number of sensing nodes increases, the error decreases significantly. In addition, the average relative error of the adaptive cubic exponential smoothing algorithm is 0.1454, while the average relative error of the adaptive cubic exponential smoothing algorithm based on the penalty error matrix is 0.0877, which is much smaller than the former. Experimental results show that the prediction effect of the improved algorithm is better than the adaptive cubic exponential smoothing algorithm.

6. Conclusion
Aiming at the problem of node energy limitation in wireless sensor networks, an adaptive cubic exponential smoothing algorithm based on penalty error matrix is used to reduce the energy loss of the network. Compared with the original algorithm, the algorithm can dynamically adjust the smoothing coefficient, so as to better adapt to the changing trend of the time series. It can be seen from the curve comparison chart that the prediction accuracy and energy saving effect have been further improved, which can effectively avoid the damage caused by excessive energy consumption.

Authors
[Profile of the author] Zhang Ping (1979-), female, Lanzhou, Gansu, associate professor, PhD, research on modeling and control of complex systems;
Liu Meiqing (1994-), female, Lanzhou, Gansu, master's degree, research on solar photovoltaic power generation system. [Profile of the author] Zhang Ping (1979-), female, Lanzhou, Gansu, associate professor, PhD, research on modeling and control of complex systems;
Liu Meiqing (1994-), female, Lanzhou, Gansu, master's degree, research on solar photovoltaic power generation system.
References

[1] Visalakshi A, Rajesh A. Implementation of an efficient extreme learning machine for node localization in unmanned aerial vehicle assisted wireless sensor networks. 2020, 33(10):n/a-n/a.

[2] Li W.Z, Tu X.M. Quality analysis of multi-sensor intrusion detection node deployment in homogeneous wireless sensor networks. 2020, 76(2):1331-1341.

[3] Aarti J, Anuj P. Ant Colony Optimization and Excess Energy Calculations Based Fast Converging Energy Efficient Routing Algorithm for WSNs. 2019, 109(6):2305-2328.

[4] Anthony M, George O, Ann K. Hybrid IEEE 802.15.6 Wireless Body Area Networks Interference Mitigation Model for High Mobility Interference Scenarios. 2018, 9(2):34-48.

[5] BAE J, SONG K, LEE H, et al. A 0.24-nJ/b Wireless Body-area-network Transceiver With Scalable Double-FSK Modulation[J]. IEEE Journal of Solid-state Circuits, 2011, 47(1): 310-322

[6] Ke H, Bichuan S. Design of Port Communication Signal Management System Based on ZigBee. 2020, 103(sp1):735-738.

[7] Christopher M. Rondeau, J. Addison Betances, Michael A. Temple, et al. Securing ZigBee Commercial Communications Using Constellation Based Distinct Native Attribute Fingerprinting. 2018, 2018

[8] Sun M, Shafia S, Atiqul I, et al. Optimization of ZigBee Network Parameters for the Improvement of Quality of Service. 2018, 6(6):1-14.

[9] PENG Y, WANG X, GUO L, et al. An Efficient Network Coding-based Fault-tolerant Mechanism in WBAN for Smart Healthcare Monitoring Systems[J]. Applied Sciences, 2017, 7(8): 817.

[10] KHATUN F, HEYWOOD A E, HANIFI S M A, et al. Gender Differentials in Readiness and Use of MHealth Services in a Rural Area of bangladesh[J]. BMC Health Services Research, 2017, 17(1): 573.

[11] Daniele R, Gisely C.D. M, José M.H.C, et al. Correction to: Use of a NAT-based assay to improve the surveillance system and prevent transfusion-transmitted malaria in blood banks. 2020, 19(1)

[12] Overview and Methods for the Youth Risk Behavior Surveillance System - United States, 2019.. 2020, 69(1):11-10.

[13] Nader M, Haleh A, Davoud K.Z. Injury surveillance information system: A review of the system requirements. 2020, 23(3):168-175.

[14] Li X.T, Howard H. Chang, Q. C, et al. A Spatial Hierarchical Model for Integrating and Bias-Correcting Data from Passive and Active Disease Surveillance Systems. 2020,

[15] Introduction of Taiwan Antibiotic Resistance Surveillance System. 2019, 35(3)