International Journal of Control and Automation
Vol. 9, No. 7 (2016), pp.79-92
http//dx.doi.org/10.14257/ijca.2016.9.7.08

A Multi-Hop Clustering Fusion Routing on Water Environment Wireless Sensor Network

Zhi-Gui Lin[1,2*], Feng-Ru Wang[1,2], Ying-Ping Liu[3, 4], Xu-Lei An[1,2] and Cai-Xia Zhang[1,2]

[1]*School of Electronics and Information Engineering, Tianjin Polytechnic University, Tianjin 300387, China*
[2]*Tianjin Key Laboratory of Optoelectronic Detection Technology and Systems, Tianjin Polytechnic University, Tianjin 300387, China*
[3]*School of Mechanical Engineering, Tianjin Polytechnic University, Tianjin 300387, China*
[4]*Tianjin City Key Lab of Modem Mechatronics Equipment Technology, Tianjin Polytechnic University, Tianjin 300387, China*
**linzhigui@tjpu.edu.cn*

Abstract

Sensor technology has been used in water environment, which comes into being a water environment wireless sensor monitoring network. Monitoring data in the network slowly change, so we propose a geographical energy-efficient multi-hop clustering fusion routing algorithm based on multilayer perceptron (MLP-GEEMHCFR) in this paper to reduce transmittingdata and save the network energy.The algorithm is implemented by initialization phase, stabilization phase and data fusion phase. A multilayer perceptron (MLP) neuronal function model is designed in the data fusion stage using the MLP, and effect of neuron thresholds on the algorithm fusion efficiency is analyzed in detail. The simulation results show that the performance of the MLP-GEEMHCFR is better than a geographical energy-efficient multi-hop clustering routing (GEEMHCR) algorithm without fusion. The data fusion efficiency increases with the neuron threshold.But when the neuron threshold is up to 1 or more, the increasing trend becomesslowly.

***Keywords**: clustering routing; data fusion; WSN; water environment*

1. Introduction

Water environment wireless sensor network (WSN) is consisted of a large number of sensor nodes (about several hundred to several thousand) through self-organization, which can acquire water environment data. TheWSN provides a new means to monitor water environment in real time. In the WSN, the data is acquired by the sensor nodes and transmitted between them through a routing protocol that is different from that of wireless network. The WSN is energy-limited, while the wireless network is not. Therefore, it is important to design an effective energy-saving routing protocol as the environmental monitoring data are large and change slowly.

Reduce transmitted data by data fusion and then extend the lifecycle of the wireless sensor networks, which is taken into account in related fields. Yao et al.[1] selected a cluster head based on characteristics of network environment and the distance between the cluster head and the base station, and reduced the amount of data by data fusion in the cluster head. Lu et al. [2] found the optimal route path and fusion placements for a given data fusion tree. Consider Mobile Agent (MA) in multi-hop environments and adopt the gradient of Bayes sequential estimation to dispatch the MA by designing the data packet and table with the specific structure, Cui et al. [3]proposed a new MA data fusion algorithm based on Bayes estimation for wireless sensor networks. MA performed data

ISSN: 2005-4297 IJCA

processing and makes data aggregation decisions at nodes rather than bringing data back to sink node, and the redundant sensory data would be eliminated. Wu et al. [4] introduced artificial intelligence and based on ant colony optimization algorithm (ACOA) and Artificial fish swarm algorithm (AFSA), and then proposed ACOA-AFSA fusion routing algorithm for underwater wireless sensor network. Depth of the water is shallow and change of the water quality in the depth direction is little in inland lakes or coastal area, where we establishes a water wireless sensor network based on radio communication to the communication delay on acoustic communication. Yang et al. [5] proposed an energy-saving routing protocol which is suitable for ocean environment. The protocol focused on the energy distribution, but there is nosignificant improvement on reducing the lifecycle of the network. Li et al. [6] applied the Low Energy Adaptive Clustering Hierarchy (LEACH) to the water environment WSN, and proposed the LEACH routing protocol based on the ideal frequency of cluster head occurrence. The energy consumption could be reduced by reasonable selecting the cluster head, but it has poor operability and limited ability to reduce energy consumption. Combined with the characteristics of wireless monitoring data in water environment, Liu et al. [7] proposed a routing algorithm of wireless energy control in water environment based on the LEACH. The algorithm selects cluster heads in successive selection mode to increase the lifecycle of the system, but the mode consumes too much time and energy.

Because monitoring data change slowly, or even doing no-change in a few hours in water environmental monitoring, we will consume more energy if the massive identical data are transmitted without processing. In addition, a water environment region is simultaneously monitored by multiple nodes, which makes some data duplicated. Thereby, it will save a lot of energy and extend the network lifecycle by data fusion before transmission. Multilayer perceptron (MLP) neural network is widely applied in wireless sensor networks to fuse data for it has advantages of fast converges and simple structure.Guo et al. [8] proposed fusion routing algorithm based on the MLP to fuse data of mobile networks.In this paper, we design a three-layer MLP neural function model for data fusion and propose a geographical energy-efficient multi-hop clustering fusion routing algorithm based on the MLP (MLP-GEEMHCFR), and analyze in detail effect of the neuron thresholds on the fusion efficiency of the MLP-GEEMHCFR algorithm.

The structure of the paper is as follows. Section 2 discusses a geographical energy-efficient multi-hop clustering routing (GEEMHCR) algorithm. Section 3 describes the theory and designs of the MLP-GEEMHCFR algorithm, and analyzes its performance. The MLP-GEEMHCFR and the GEEMHCR algorithms are compared and analyzed in Section 4. Finally, Section 5 summarizes and concludes.

2. The GEEMHCR Algorithm

Considering the size of wireless monitoring network region and the number of the nodes in the network, we firstly divide the network into several square areas by the GEEMHCR algorithm [9], and then select the cluster head. The residual energy of each node is considered in the process of cluster head selection. The cluster-head nodes communicate with the base station using the protocol of multi-hop in the process of data transmission. The GEEMHCR algorithm is implemented by initialization phase and stabilization phase. The initialization phase completes clustering, selection of cluster-head nodes and establishment of the routing, while the stabilization phase is mainly for routing maintenance.

2.1. Initialization Phase

A cartesiancoordinate is firstly established in the monitored area, and then nodes are clustered. The radio model indicates that the energy consumption of a sensor node sending k bits to d distance($E_{send}(k,d)$) is shown as Equation (1).

$$E_{send}(k,d) = E_{s\_elec}(k) + E_{amp}(k,d) = kE_{elec} + k\varepsilon_{amp}d^4 \quad (1)$$

where, E_{s_elec} is the energy consumption tosend *k* bits of data, $E_{amp}(k,d)$ is the energy consumption amplifying k bits of data to transfer to d distance, E_{elec} is the energy consumption to send or receive 1 bit of data, *d* is the distance of signal transmission, *k* is data bit numbers to be sent and ε_{amp} is the magnificationof signal,

$\varepsilon_{amp} = \varepsilon f_s = 0.0013 \ pJ / bit / m^4$.

The energy consumption of a sensor node receives *k* bit data ($E_{receive}(k)$) is shown as Equation (2).

$$E_{receive}(k) = E_{r\_elec}(k) = kE_{elec} \quad (2)$$

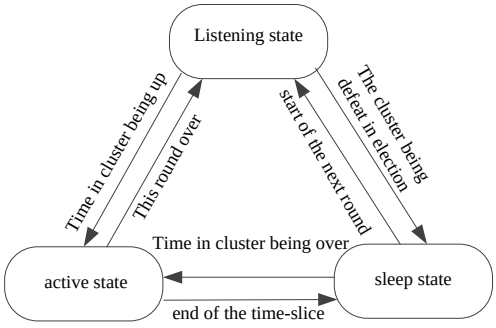
where, E_{r_elec} is the energy consumption to receive k bit of data, E_{elec} is the energy consumption to send or receive 1 bit of data.

To a monitoring area, we can derive an optimal clustering number (m_{opt}) by the equation (1) and the equation (2).

$$m_{opt} = \sqrt[4]{\frac{n}{6\pi}} \sqrt[4]{\frac{\varepsilon_{fs}}{\varepsilon_{amp}}} \frac{L\sqrt{L}}{d_{toBS}} \quad (3)$$

where, n is numbers of the network nodes, L is sidelength of the monitoring region, ε_{fs} is the magnification of the signal and d_{toBS} is the average distance from a node in the monitoring area to the base station.

We select cluster heads after the establishment of clusters. Nodes have three operating states including the listening, active and sleep state. Conversion among the states is shown in Figure 1.



.

Figure 1.The Conversion between the Node States

All the sensor nodes are in listening state after clustering, and then compete with cluster head nodes. The winner enters the active state and becomes a new cluster head, otherwise, do the sleep state. After a round of stable operational state, the cluster head in the active state enters the listening state to participate in the next round of selection again. If nodes in the sleep state enter the active state in their own time slice, it would transfer to the sleep state again after the time slice. If the nodes in the sleep state, which enter the active state in their own time slice and communicate with the cluster head, would enter the listening state after a round of stable operational state to participate in cluster heads selection.

In the GEEMHCR algorithm, data transmission by multi-hop model between the base station and cluster headsare shown in Figure 2. A multi-hop path is established by multi-hop routing with the shortest path. That is, the base station(BS) broadcasts the status information of a node by using the Restricted-Flooding mode, and then cluster head nodes broadcast theirown status information after it receives the status information. This process repeats until all nodes in the network receive the status information, and then the minimum path between each cluster head node and the base station is established. The algorithm ensures that each cluster head node only communicates with the nearest neighbor cluster head node, so that it reduces the overall energy consumption of the network, and makes the node energy consumption of the whole network distributes more

evenly and extends the network lifecycle.

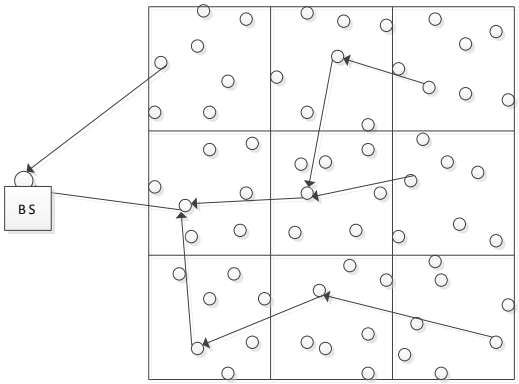


Figure 2. Structural Representation of the GEEMHCR Algorithm

1.2. Stabilization Phase

The routing is maintained in the stabilization phase. When another node takes on a cluster head and a new cluster head node is produced, the routing table maintenance is performed according to the following steps:

Step1: Old cluster head node sends its routing table to a new cluster head node.

Step2: The new cluster head node broadcasts its state information to its neighbor cluster head node.

Step3: After received the cluster head node's state information, neighbor cluster head nodes search in their own routing table according to the cluster number in the state information, and perform different process based on the search results. If neighbor cluster heads and other nodes are in the same cluster, the corresponding records in the routing table are deleted and the state information of the received node in the routing table is stored. Or, the hops of the received nodes and that of the current node are compared. If the former is less than the later, the state information is stored to the routing table, otherwise, it is dropped.

2. The MLP-GEEMHCFR Algorithm

2.1. The Algorithm Design

Compared with the GEEMHCR,the MLP-GEEMHCFR algorithm appends data fusion phase, and implements by initialization phase, stabilization phase and data fusion phase.

Considering the features of the environmental monitoring data, we use the MLP to fuse data in the data fusion phase. The MLP [10] is a typical feed-forward neural network which consists of an input layer, a number of hidden layers and an output layer. It has been proved by Robert Hecht-Nielson that a three-layer MLP, which contains a hidden layer,can make any non-linear mapping. But error of training result based on the MLP is higher and its identification precision is lower. It can improve the identification precisionand reduce errors that we increase the number of hidden layers in the MLP, which makes a network structure more complex and increase training time of the network connection weights. Currently, there is no fixed calculation method to determine the number of hidden layers. We generally do empirical formula. In the MLP, hidden layer numbers relate with many parameters, such as samples numbers, input units numbers, output units numbers and so on. According to characteristics of water environment wireless sensor networks, that is input data types, data slowly changing,and abnormal databecause of burst event and so on, and combining with training time and error of the network, the paper introduces a MLP with two hidden layers, as shown in Figure 3. Cluster members belong to the input layer and the first hidden layer, and the cluster head nodes belong to the second hidden layer and the output layer. Suppose each cluster has n

nodes, and each node monitors m kinds of water quality parameters such as pH value, temperature, dissolved oxygen, salinity and so on, so neuron numbers in the input layer are $n \times m$. But neuron numbersof the second hidden layer and the output layerhave nothing to do with node numbers in each cluster, and are decided by network environment. Neurons in the input layer do not allconnect to neurons in the first hidden layer, and it is the same between the first hidden layer and the second hidden layer in order to fuse the same typological data. The neurons in the second hidden layer allconnect to neurons in the output layer to comprehensively process different typological data.

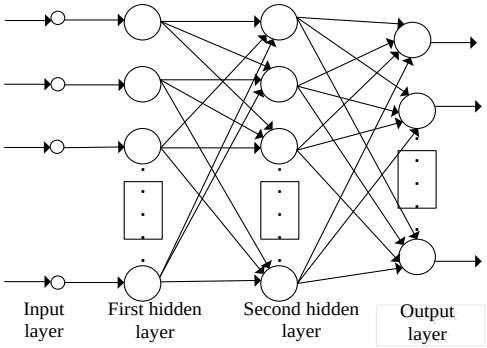


Figure 3.Model of the MLP

2.2. Design of the Neuronal Function Model

(1) Function model of the first hidden layer neuronal

We make the first hidden layer neurons lie in cluster member nodes, numbers of the first hidden layer neurons in each cluster member node are determined by types of the data gained by the sensor nodes. The function model of the first hidden layer neurons is divided into input, processing and output area, as shown in Figure 4. We use the weighted-average function to process data in input area. Each neuron in the first hidden layer has only a parameter, so the function in input area is an identical function.

$$y(t) = f(p, w) = p * w = p \quad (4)$$

where, $y(t)$ is the current value, $f(p,\omega)$ is processing function, p is an input data, ω is a weight value and equals 1.

Inprocessing area, we calculate the difference value between the current value and last stored value, as equation (5) shows, and decide whether the current value is stored by equation (6).

$$g_d(t) = g_d(y(t), g(t-1)) = |g(t-1) - y(t)| \quad (5)$$

Where, $y(t)$ is the current value, $g(t-1)$ is last stored value, $g_d(t)$ is the difference value.

$$g(t) = g(g_d(t), g(t-1), y(t)) = \begin{cases} g(t-1), & g_d(t) < \theta_1 \\ y(t), & g_d(t) \geq \theta_1 \end{cases} \quad (6)$$

Where, $y(t)$ is the current value, $g(t-1)$ is last stored value, $g(t)$ is current value being stored, and $g_d(t)$ is the difference value.

We decide whether the stored value is outputted through comparing the difference value with a threshold set. In equation (7), $a(t)$, being a standard two-valued function, is an output function of the first hidden layer neuron.

Cluster member nodes do not send the data to a cluster head if the difference between the current dataand the last stored data is approximately equal to 0, and then send the data only when the difference is greater than the threshold θ_1.The processing method avoids a lot of duplicate or similar data being sent to the cluster head node.

$$a(t) = a(g_d(t), g(t)) = \begin{cases} No\ \ output, & g_d(t) < \theta_1 \\ g(t), & g_d(t) \geq \theta_1 \end{cases} \quad (7)$$

where, $g(t)$ is the current value stored, $g_d(t)$ is the difference value.

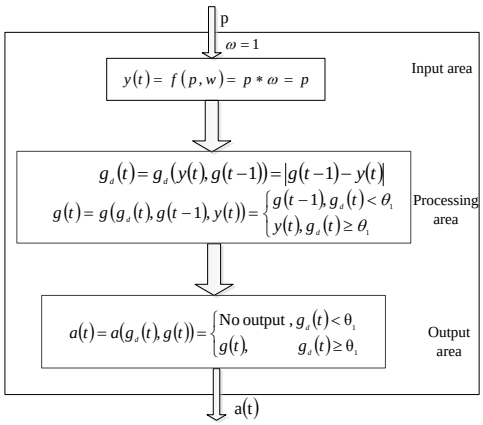


Figure 4. Neuronal Function Model of the First Hidden Layer

(2) Function model of the second hidden layer neuronal

We make second hidden layer neurons lie in cluster head nodes. The neuron numbers, function model and function of the layer are related to the characteristics of water environment monitoring data. Suppose that three parameters, such as pH value, temperature and dissolved oxygen, are monitored, and calculate the maximum, minimum and average values of each parameter within 24 hours, so the neuron numbersof the hidden layer is 9. We also divide the neuron model of the second hidden layer into input, processing and output functional areas. Taking temperature data as an example, we discuss performance of the neuron model.

The neuronal model of average temperature within 24 hours is shown in Figure 5.Suppose that the neuron has n input values $T_j(j=1\ldots n)$ and one output value T_{avg}. Because weight of each T_j is equal to 1, so we calculate directly mean value of $T_j(j=1\ldots n)$ in input area. Similarly, we also use an identityfunction in processing area and output area.

$$y(t) = f(T, w) = \frac{1}{n}\sum_{j=1}^{n} \omega_j T_j = \frac{1}{n}\sum_{j=1}^{n} T_j \quad (8)$$

$$a(t) = a(g(t)) = g(t) = g(y(t)) = y(t) \quad (9)$$

where, $y(t)$ is the current value, ω is a weight value, $g(t)$ is processing function and $a(t)$ is output function.

The cluster head only receives the temperature data sent by a cluster member node at a time, so there are only two input signals in the highest temperature neuron within 24 hours, as figure (6) shows. The two signals are current temperature and time. To simplify the calculation, the 24 hours are divided into four timeperiodes as: 09:00 am-15:00 pm and 15:00 pm-21:00 pm on the day before, 21:00 pm-03:00 am and 03:00 am-09:00am on the day. Each neuron saves five temperature values, that is $g_1(t)$, $g_2(t)$, $g_3(t)$, $g_4(t)$ and $g_5(t)$, which corresponds to the maximum temperatures before 24 hours, 18 hours, 12 hours, 6 hours and the current time, respectively. Suppose the current time is 09:00 am-15:00 pm.

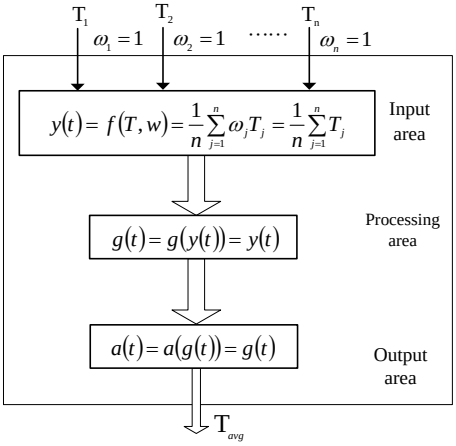


Figure 5. The Function Model of Average Temperature Neuronal Within 24 Hours.

In input area, we process input data by three functions. The first function is a weighted sum function. The function transforms into an identity function because input data has only a temperature data.

$$y_1(t) = f_1(T_j, w) = T_j \omega_j = T_j \quad (10)$$

where , T_j is a temperature value gained by the *j*th member node, ω_j is a weight value.

The second function in input area determines the time period of the current time. In equation (11), The time period ($y_2(t)$)of 21:00pm-03:00am,03:00am-09:00am, 09:00am-15:00pm and 09:00am-15:00pm corresponds to 0,1,2,3 respectively.

$$y_2(t) = f_2(Time_{now}) = rounddown\left\{\frac{(Hour(Time_{now}) + 3) \bmod 24}{6}\right\} \quad (11)$$

The third function in input area judges whether the current time enters sequentially every time period.

$$y_3(t) = f_3(y_2(t), y_2(t-1)) = \begin{cases} 1, & y_2(t) \neq y_2(t-1) \\ 0, & y_2(t) = y_2(t-1) \end{cases} \quad (12)$$

If timeperiode, to whichthe current timebelongs, is consistent with that of last system time, $y_3(t)$ is equal to 0 .Or $y_3(t)$ is equal to 1.

In processing area, we calculate the highest temperatures of the five time period in 24 hours respectively, that is $g_1(t)$, $g_2(t)$, $g_3(t)$, $g_4(t)$ and $g_5(t)$, see equation (13) and equation (14).

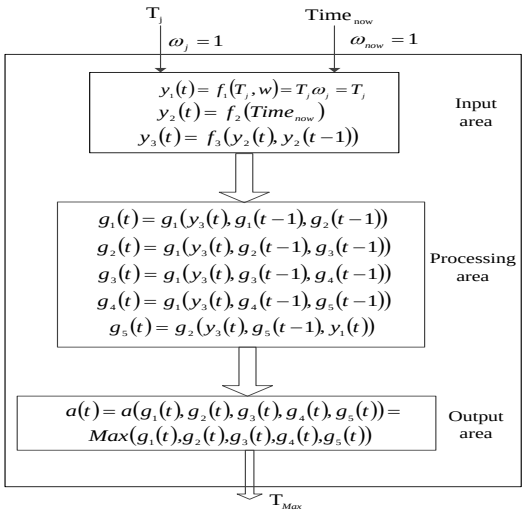


Figure 6. The Function Model of the Highest Temperature Neuronal within 24 Hours

$$g_i(t) = g_1(y_3(t), g_i(t-1), g_{i+1}(t-1)) = \begin{cases} g_i(t-1), y_3(t) = 0 \\ g_{i+1}(t-1), y_3(t) = 1 \end{cases} \quad (i = 1,2,\ldots,4) \quad (13)$$

In equation (13), if $y_3(t) = 0$, the time period that the clocks belong and corresponding highest temperature values are not change, respectively. If $y_3(t) = 1$, the time periods that the clocks belong are not change, and corresponding highest temperature values are that of next time period, respectively.

$$g_5(t) = g_2(y_3(t), g_5(t-1), y_1(t)) = \begin{cases} y_1(t), & y_3(t) = 1 \\ g_5(t-1), y_3(t) = 0 \text{ and } g_5(t-1) \geq y_1(t) \\ y_1(t), & y_3(t) = 0 \text{ and } g_5(t-1) < y_1(t) \end{cases} \quad (14)$$

In equation (14), if $y_3(t) = 1$, the system clock enters the next time period and the corresponding highest temperature data is lately received value $y_1(t)$. If $y_3(t) = 0$, the time period that the system clock belong does not change, and the corresponding maximum temperature data is the larger value between the lastly received value and the conserved value.

In output area, we calculate and output the maximum temperature value in the five time periods by equation (15), which is also the maximum value within 24 hours.

$$a(t) = a(g_1(t), g_2(t), g_3(t), g_4(t), g_5(t)) = Max(g_1(t), g_2(t), g_3(t), g_4(t), g_5(t)) \quad (15)$$

Similarly, it is same with the average temperature that we can calculate the average values of the pH value and dissolved oxygen within 24 hours in the second hidden layer, and with the maximum temperature within 24 hours that we can calculate the minimum temperature value, the maximum and minimum pH values, and the maximum and minimum dissolved oxygen values within 24 hours.

(3) Function model of the output layer neuronal

We make output layer neurons lie in cluster head nodes. The function model of the output layer neuron is also divided into input, processing and output area, as shown in Figure 7.

The function model is similar to that of the first hidden layer, but their difference is mainly that the former is multi-input,and then the later is single-input.

$$y(t) = f(I, w) = \sum_{j=1}^{m} w_j I_j \quad (16)$$

Where, I_j is output of the *j*th neuron, ω_j is a weight value.

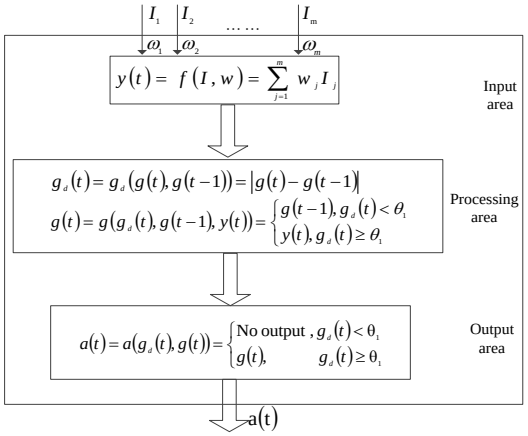


Figure 7. Function Model of the Output Layer Neuronal

2.3. Performance Analysis

(1)The first hidden layer

Suppose the water environment data acquisition period is t hours, there are $1/t$ data samples acquired per hour, and the data changes in δ linear. We can calculate output samples numbers x processed in the first hidden layer by equation (17).

$$x = \begin{cases} 1/t, & \theta_1 = 0 \\ 0, & \theta_1 > 0 \text{ and } \delta = 0 \\ \frac{1}{t} \times \frac{|\delta|}{\theta_1}, & \theta_1 > 0 \text{ and } \delta \neq 0 \end{cases} \tag{17}$$

Where, θ_1 is threshold value of the first hidden layer neuron. According to Equation (17), output data numbers of the neurons in the layer are determined by $|\delta|$ and θ_1, which decrease with the increase of θ_1 and increase with the increase of $|\delta|$. The ratio of the output sample numbers to the input sample numbers η_1 is regarded as the data fusion rate for the layer.

$$\eta_1 = \begin{cases} 1, & \theta_1 = 0 \\ 0, & \theta_1 > 0 \text{ and } \delta = 0 \\ \frac{|\delta|}{\theta_1}, & \theta_1 > 0 \text{ and } \delta \neq 0 \end{cases} \tag{18}$$

During the monitoring process of water environment, the acquired data changes according to the environment. Therefore, θ_1 could be adjusted according to environment. If we increase θ_1, the amount of data transmitted from the cluster member nodes to the cluster head decreases, and then the energy consumption also decreases. However, θ_1 affects data processing, it should not be increased in order to reduce energy consumption, so an appropriate threshold should be chosen.

(2) The second hidden layer

We can calculate the fusion rate in the second hidden layer η_2 by equation (19).

$$\eta_2 = \frac{\sum_{i=1}^{m} s_i}{n \times m} \tag{19}$$

where, n are cluster node numbers, m is numbers of the acquired data type and s_i are neuronal numbers of the second hidden layer processing the i th data. As Equation (19) shows, the fusion rate η_2 increases with the decrease of n and the increase of $\sum_{i=1}^{m} s_i$.

(3) Neurons of the output layer

The neuronal model of the output layer is similar to that of the first hidden layer, as shown in Equation (20).

$$\eta_3 = \begin{cases} \dfrac{1}{m}, \theta_2^j = 0 \\ \sum_{i=1} s_i \\ 0, \quad \theta_2^j > 0 \text{ and } \delta_y^j = 0 \\ \dfrac{\left|\delta_y^j\right|}{\theta_2^j}, \ \theta_2^j > 0 \text{ and } \delta_y^j \neq 0 \end{cases} \tag{20}$$

where δ_y^j is exchange value of processing results of the j th neuron in the input area y, δ_2^j is the threshold of the j th neuron in output layer. The fusion rate η_3 increases with the threshold.

If δ_y^j is equal to δ_2^j, data fusion rate of the output layer neuron η_4 is shown as Equation (21).

$$\eta_4 = \begin{cases} \dfrac{k}{n \times m}, \theta_2 = 0 \\ 0, \quad \theta_2 > 0 \text{ and } \delta_p = 0 \\ \sum_{i=1}^{m} s_i \times \dfrac{\left|\delta_p\right|}{\theta_2} \times \dfrac{k}{n \times m}, \ \theta_2 > 0 \text{ and } \delta_p \neq 0 \end{cases} \tag{21}$$

where, k is numbers of the output layer neurons, θ_2 is equal to $\theta_2^1 (= \theta_2^2 = \ldots = \theta_2^k)$.

3. Simulation and Analysis

We use the NS-2 tool for simulation analysis of the MLP-GEEMHCFR algorithm and set monitoring area as100m × 100m, and100 nodes are uniformly distributed in the area. The coordinates of the base station is (0, 50).

3.1. Analysis of Performance

We compare and analyze the MLP-GEEMHCFR and the GEEMHCR from three respects including the amount of surviving node, the total energy consumption of the network and the amount of data received by the base station, and illustrate difference between the algorithms.

(1)The amount of surviving nodes

Changing trends of the surviving nodes versus time are compared in the two algorithms, as shown in Figure 8. Network lifecycle of the MLP-GEEMHCFR algorithm and the GEEMHCR algorithm are 750s and 680s, respectively, which shows that data fusion extends nearly10% of the network lifetime.

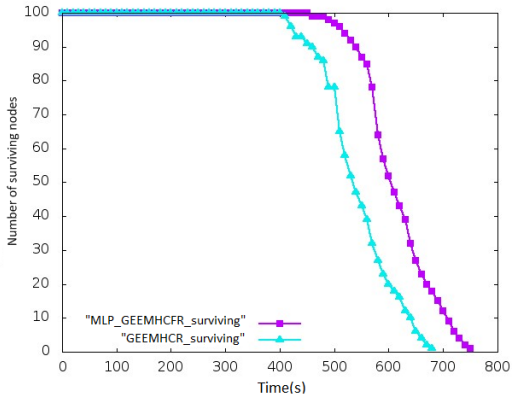


Figure 8. Changing Trends of the Surviving Nodes Versus Time

(2)The total energy consumption

Changing trends of the total energy consumption versus time on the two algorithms are compared, as shown in Figure 9. The total energy consumption of the MLP-GEEMHCFR algorithm is obviously lower than that of the GEEMHCR algorithm before 400s, while the former is significantly lower than the latter in the last 400s. All energy of the network at t=700s is exhausted when using the GEEMHCR algorithm. While exhausting time of the network energy is 760s when using the MLP-GEEMHCFR algorithm. This indicates that the MLP-GEEMHCFR algorithm is better than the GEEMHCR in reducing the network energy consumption.

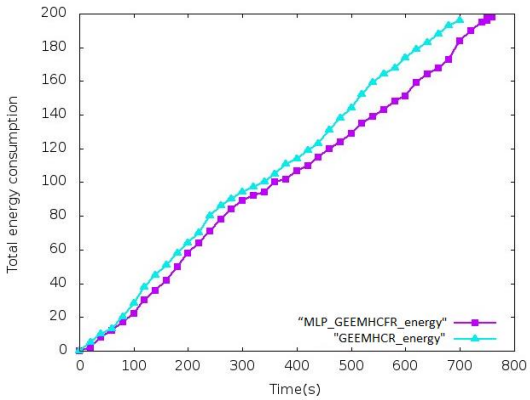


Figure 9. Changing Trends of the Total Energy Consumption Versus Time

(3)The amount of data received by base station

Changing trends of the amount of data received by the base station versus time on the two algorithms are compared, as shown in Figure 10. When the simulation using the GEEMHCR algorithm ends at t=700s, numbers of data packets transmitted from all network nodes to the base station are 78000. While it using the MLP-GEEMHCFR algorithm ends at t=760s, and data packet numbers are 100000. Thus, the MLP-GEEMHCFR algorithm is better than the GEEMHCR algorithm in receiving the data packets.

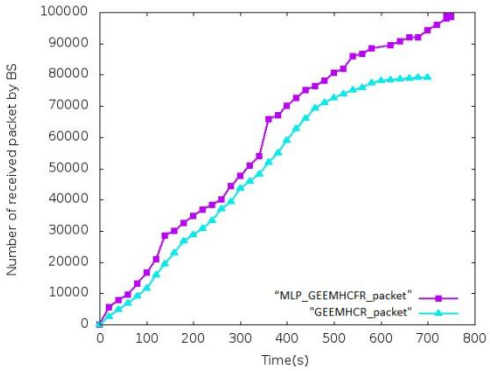


Figure 10. Changing Trends of the Amount of Data Received By the Base Station versus Time

3.2. The Effect of Main Parameters on the MLP-GEEMHCFR Algorithm

In this section, we mainly analyze effect of the first and second hidden layer thresholds θ_1 and θ_2 on the fusion rate of the MLP-GEEMHCFR algorithm. We set the θ_1 and θ_2 as (0,0),(0,0.5),(0.5,0),(0.5,0.5),(1,1),(1.2,1.2), which are sent to each sensor node by the base station, respectively, and compare the effect in the aspects of the number of survival nodes, the total energy consumption and the amount of data received by base station.

(1)The number of surviving nodes

The effect of the thresholds θ_1 and θ_2 on the number of survival nodes in water environment wireless sensor network is shown in Figure 11. At the same time, numbers of surviving nodes are the fewest when the threshold is (0, 0). The number and the lifecycle of the network increase with the increase of the threshold,*e.g.*, the lifecycle of the network is 680s when $\theta_1 = 0$ and $\theta_2 = 0$ and reaches the maximum of 785s when $\theta_1 = 1.2$ and $\theta_2 = 1.2$.

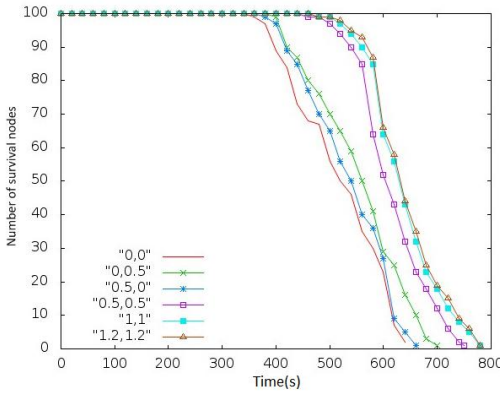


Figure 11. Effect of the Thresholds θ_1 and θ_2 on the Number of Survival Nodes

(2)The total network energy consumption

The effect of the thresholds θ_1 and θ_2 on the total energy consumption of Water Environment Wireless Sensor Network is shown in Figure 12. At the same time, the total network energy consumption reaches the maximum and the lifecycle is the minimum when $\theta_1 = 0$ and $\theta_2 = 0$ .With the increase of θ_1 and θ_2 , energy consumption speed in the network gradually slows and the lifecycle of the network increases, *i.e.* the fusion rate of the MLP-GEEMHCFR algorithm is improved. The total energy consumption and the lifecycle at $\theta_1 = 1.2$, $\theta_2 = 1.2$ is almost equal to those at $\theta_1 = 1$, $\theta_2 = 1$, which indicates that values of θ_1 and θ_2 have less effects on the total energy consumption when $\theta_1 > 1$,

$\theta_2 > 1$. So the thresholds have almost no effect on the data fusion rate of the MLP-GEEMHCFR algorithm when θ_1 and θ_2 are up to certain values.

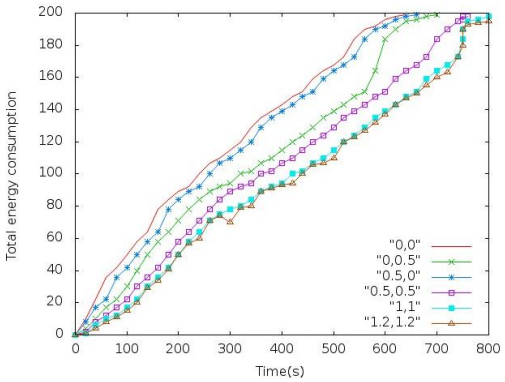


Figure 12. Effect of the Thresholds θ_1 and θ_2 on the Total Energy Consumption

(3) The amount of data received by base station

Effect of the thresholds θ_1 and θ_2 on the amount of data received by the base station in the water environment wireless sensor network is shown in Figure 13. At the same time, the data sent to the base station decrease with the increase of the thresholds θ_1 and θ_2, so data received by the base station is the maximum when $\theta_1 = 0$, $\theta_2 = 0$. From Figure 13, we see the number of survival nodes increase with increase of the thresholds θ_1 and θ_2, so the sizes of the Acquisition data also increase in the next time. This complementary function makes changes of thresholds θ_1 and θ_1 having less effect on the amount of data received by the base station. When $\theta_1 > 1$, $\theta_2 > 1$, θ_1 and θ_2 almost have no effect on the amount of data received by the base station, which shows that the thresholds have no obvious effect on the fusion rate of the MLP-GEEMHCFR algorithm when threshold θ_1 and θ_2 are up to certain values.

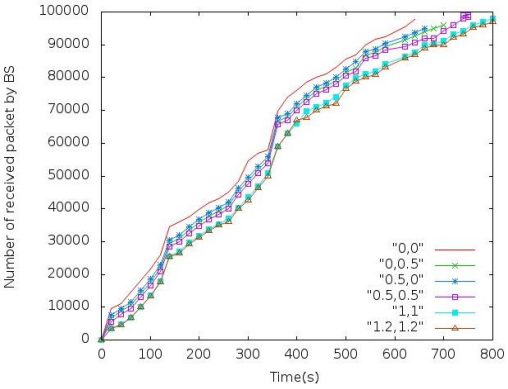


Figure 13. Effect of the Thresholds θ_1 and θ_2 on the amount of Data Received by the Base Station

4. Conclusions

We propose a geographical energy-efficient multi-hop clustering fusion routing algorithm based on multilayer perceptron (MLP-GEEMHCFR) for the feature that wireless monitoring data change slowly in the water environment in the paper. The algorithm is implemented by initialization phase, stabilization phase and data fusion phase. The initialization phase mainly completes clustering, selection of cluster-head nodes and establishment of the routing. In the process of cluster head nodes selection, the residual energy and times of a node selected as a cluster head are all considered to avoid the

energy consumption unbalance. The stabilization phase maintains routing. The fusion phase uses the MLP to deal with data fusion to reduce the amount of data transferred and save energy.

Cluster members are assigned to the input layer and the first hidden layer, while cluster head nodes are assigned to the second hidden layer and the output layer. We design the MLP neuronal function model according to the feature of water environment monitoring data, and analyze the effects of the thresholds θ_1 and θ_2 on fusion rate of the MLP-GEEMHCFR algorithm. The analysis results indicate that the amount of output data decease, the fusion rate improves and the accuracy decreases with the increase of the thresholds θ_1 and θ_2 .

On the NS2 simulation platform, for a 100m×100m water environment monitoring area, we analyze the MLP-GEEMHCFR algorithm from the aspects of survival nodes in the network, the total energy consumption and the amount of data received by the base station. The analysis results show that the performance of the MLP-GEEMHCFR algorithm is obviously better than the GEEMHCR algorithm. The fusion rate increases with the increases of the thresholds θ_1 and θ_2 , but effect on the fusion rate is little when the threshold θ_1 and θ_2 are greater than 1. We select the thresholds θ_1 and θ_2 according to the monitoring environment and the scale of the network for different water environment.

Acknowledgments

This study is financially supported by the National Natural Science Foundation of China, 2013 (Project No. 61372011)

References

[1] X.-B. Yao, X.-D. Wang and Sh.-J. Li, "A new data fusion routing algorithm for wireless sensor networks", 2011 IEEE International Conference on Computer Science and Automation Engineering , Shanghai, China, **(2011)**, pp. 676-680.
[2] Z.-Q. Lu, S. L. Tan and J. Biswas, "D2F: A Routing Protocol for Distributed Data Fusion in Wireless Sensor Networks", Wireless Personal Communications, vol. 70, no. 1, **(2013)**, pp. 391-410.
[3] Z.-W. Cui; Y. Zhao, D.-Y. Xu and Y.-A. Zuo, "Dynamic Routing Based on Bayes Estimation for Wireless Sensor Networks", Journal of Networks, vol. 8, no. 6, **(2013)**, pp. 1403-1410.
[4] . H.-F. Wu, X.-Q. Chen, C. J. Shi, Y.-J. Xiao and M. Xu, "An ACOA-AFSA Fusion Routing Algorithm for Underwater Wireless Sensor Network", International Journal of Distributed Sensor Networks, **(2012)**, pp. 1-9.
[5] D.-H. Yang and J. Wang, "A adaptive Wireless Sensor Networks for Marine Monitoring", Advanced Materials Research, vol. 542-543, **(2012)**, pp. 169-178.
[6] C.-M. Li. G.-P. Tan and J.-Y. Wu, "Analyzing Cluster-head Selection Mechanisms and Improving the LEACH", 2011 International Conference on Electronics, Communications and Control. Ningbo, China, **(2011)**, pp. 747-750.
[7] Y.-P. Liu, B.-W. Xiao and Y. A. Liu, "Energy controlling route protocol based on LEACH in water environment wireless network", Control Engineering of China. In press.
[8] Z.-H. Guo, S. Sheikh, C. Al-Najjar, K. Hyun and B. Malakooti, "Mobile adhoc network proactive routing with delay prediction using neural network", Wireless Networks, vol. 16, no. 6, **(2010)**, pp. 1601-1620.
[9] Z.-G. Lin, Z.-X.Cheng and Y.-P. Liu, "A Geographical Energy-efficient Multi-Hop Clustering Rounting", Application Research of Computers, in press, **(2015)**.
[10] L.-Y. Li, N. Wang and W. Zhang, "Neural-network Based Aggregation Framework for Wireless Sensor Networks", Computer Science, vol. 35, no. 12, **(2008)**, pp. 43-47.

