Neural Comput & Applic (2013) 23:1999–2013
DOI 10.1007/s00521-012-1248-z

ISNN2012

DLRDG: distributed linear regression-based hierarchical data gathering framework in wireless sensor network

Xin Song · Cuirong Wang · Jing Gao ·
Xi Hu

Received: 26 April 2012 / Accepted: 19 October 2012 / Published online: 5 November 2012

Abstract For many applications in wireless sensor network (WSN), the gathering of the holistic sensor measurements is difficult due to stringent constraint on network resources, frequent link, indeterminate variations in sensor readings, and node failures. As such, sensory data extraction and prediction technique emerge to exploit the spatio-temporal correlation of measurements and represent samples of the true state of the monitoring area at a minimal communication cost. In this paper, we present DLRDG strategy, a distributed linear regression-based data gathering framework in clustered WSNs. The framework can realize the approximate representation of original sensory data by less than a prespecified threshold while significantly reducing the communication energy requirements. Cluster-head (CH) nodes in WSN maintain linear regression model and use historical sensory data to perform estimation of the actual monitoring measurements. Rather than transmitting original measurements to sink node, CH nodes communicate constraints on the model parameters. Relying on the linear regression model, we improved the CH node function of representative EADEEG (an energy-aware data gathering protocol for WSNs) protocol for estimating the energy consumption of the proposed strategy, under specific settings. The theoretical analysis and experimental results show that the proposed framework can implement sensory data prediction and extracting with tolerable error bound. Furthermore, the designed framework can achieve more energy savings than other schemes and maintain the satisfactory fault identification rate on case of occurrence of the mutation sensor readings.

Keywords Wireless sensor network ·
Data gathering strategy · Distributed linear regression ·
Energy efficient

X. Song (✉) · C. Wang · J. Gao · X. Hu
School of Information Science and Engineering,
Northeastern University, 110819 Shenyang, China
e-mail: bravesong@163.com

X. Song
The Key Laboratory of Complex System and Intelligence
Science, Institute of Automation, Chinese Academy of Sciences,
100190 Beijing, China

1 Introduction

In recent dramatic development in low-power embedded wireless communication devices, digital electronic technology has made possible scenarios in which thousands of sensor nodes are seamlessly embedded in physical world and use self-organization method to form a wireless sensor network (WSN). Their wide range of applications is based on the possible use of various sensor types (e.g., thermal, visual, acoustic, seismic, magnetic, radar, vibrancy, etc.) in order to monitor a wide variety of environment conditions (e.g., temperature, humidity, illumination, pressure, object presence and movement, noise levels, etc.). However, one of the most severe limitations of sensor devices is their limited energy supply and one of the most crucial goals in designing efficient monitoring systems in WSN is minimizing energy consumption in the network. Energy consumption depends on many factors, such as deployed geography location, system configuration, device property, which may differ for different application in WSN. Because of the complex nature of a WSN, energy-efficient protocols, algorithm designed for all layers of the networking infrastructure are critically important for energy conservation besides impact on actual circumstance. In the Rivest, Shamir, and Adleman (RSA) encryption experiment in Ref. [1], for the same volume of data, the energy consumption of computation with node microprocessor is less than 1 % of that for transmission with node radio module. It is worth noting that there have been evidences showing that energy consumption due to computation is insignificant, compared with the communication cost [2]. In-network data aggregation (e.g., computing average of some values) is attractive since the amount of data can be reduced. Unfortunately, data aggregation can lose much of the feature structure in the sensory data, providing only coarse statistics information. It is noteworthy that the data prediction technology is the other one energy-efficient approach, which builds an extraction model of monitoring data to perform data evolution prediction. It is mainly aimed at reducing the energy consumption by the radio communication subsystem at intermediate nodes between the sources and the sink. The strategy not only can extract much more complete structure feature of the sensory data than most aggregation schemes but also use less communication energy than methods that gather all reading for every sensor node. The existing data prediction models for WSN have achieved the energy-efficient monitoring based on the observation that the sensor nodes capable of local computation implement the possibility of model training and updating in a distributed way. However, the integrated and energy-efficient data gathering framework combined clustering technology with prediction model which conjoins fault-tolerant strategy was not proposed and discussed.

In this paper, we proposed and evaluated a distributed linear regression-based hierarchical date gathering framework in WSN, namely, the distributed linear regression-based data gathering (DLRDG). DLRDG is a cluster-based hierarchical structure and uses the concept of approximation based on linear regression by less than a prespecified threshold. The form process takes the spatio-temporal correlation sufficiently into account when the sensed data from the environment is the generally linear tendency. By forwarding the coefficients of the basis regression model between cluster-head (CH) nodes and sink, the data gathering framework can represent the structure of the sensed data while decreasing the passing volumes of monitoring data for saving node energy. On the other hand, we discussed the model update and fault-tolerant strategy when the mutation of the sensed data occurred in the WSN monitoring process.

The primary contributions of the paper are summarized as follows:

1. In cluster-based hierarchical WSN, we proposed a new linear regression-based prediction approximation strategy that exploits correlations both within the value of periodicity as well as among values of quantities (e.g., pressure, temperature, and humidity).
2. We present a cluster-head nodes tree-based communication topologies that can effectively support the transmit operation of the regression model parameters in WSN.
3. We proposed a statistical hypothesis testing-based fault-tolerant strategy for linear regression model when the mutation of the sensed data occured. The reliability of sensor node was identified by inter-comparison of the matching degree between local data reading and statistical characteristics.
4. We present an evaluation of DLRDG framework using WSN deploy at gymnasium in miniature and NS2 simulator, demonstrating the accuracy of the linear regression model and analyzing the energy consumption.

The rest of this paper is organized as follows: in Sect. 2, we briefly review some closely related works. Section 3 presents WSN model and assumptions. The proposed data gathering framework DLRDG is derived and discussed in Sect. 4. The validity analysis and performance evaluation of the DLRDG is presented in Sect. 5. Finally, the conclusions and future work directions are described in Sect. 6.

2 Related work

In recent years, WSN has gained increasing attention from both the research community and actual users. Typically, a sensor node in WSN is a tiny device that includes three components: a sensing subsystem for data acquisition from the physical environment, a processing subsystem for local data storage and essential computing, and a wireless communication subsystem for data transmission [3]. Sensor nodes may be constrained in a limited power battery and deployed in a hostile or unpractical environment. Thus, in data-centric WSN, energy consumption is one of the most important factors to be considered in designing data gathering protocol and in-network data processing algorithms. It has been well testified that the data transmission is the most energy consuming among all operations of a sensor device. Many approaches for energy-efficient monitoring have been explored to reduce the volume of in-network data transmission, such as data aggregation. The reduction in data communication through aggregation is attractive since extraction of holistic sensor measurements can be unnecessarily requiring large amounts of communication consumption that drains the constrained energy of sensor devices. Reference [4] considered the problem of designing a distributed schedule for data aggregation from networks to sink node with minimum time slot delay. The algorithm is a nearly constant approximate strategy, which significantly reduces the aggregation delay in multihop WSN. Sensor nodes consume different energy in different radio states (transmitting, receiving, listening, sleeping, and being idle). So, a key challenging question in WSN is to schedule nodes' activities to reduce energy consumption.

Efficient scheduling for data collection and aggregation has been extensively studied recently for sensor network [5, 6]. Besides, the optimization techniques were introduced into optimal data gathering in WSN. The goal of the research work is to minimize the data gathering latency and at the same time balance the energy consumption among the nodes, so as to maximize the network lifetime [7–11]. In a cluster-based network, however, each cluster covers a small number of sensor nodes within a smaller local range of the network. This makes it more feasible to locally apply distributed source-coding technique within each cluster. Slepian–Wolf coding technique can completely remove data redundancy with no need for inter-nodes communication; therefore, distributed data aggregation using Slepian–Wolf coding for reducing possible correlation in the data generated between different clusters was proposed [12]. Because of the strictly limited sensor nodes energy, the use of mobile agent has been suggested as an intrinsically distributed computing technology in the field of data aggregation for WSN. The local data of an sensor node can be combined with the data collected by an mobile agent from other nodes in a way that depends on the specific program code of the mobile agent so that the total data volume can be reduced [13]. The more powerful mobile agent is better suited to serve as a CH to perform more tasks [14]. It is difficult that the all sensor data were accurately extracted and gathered. Reference [15] proposed data-aggregation techniques based on statistical information extraction that capture the effects of aggregation over different scales. An accurate estimation of the distribution parameters of sensory data using the expectation–maximization algorithm was designed. Some research teams proposed a data reduction algorithm for the dissemination of historical measurements in constraint sensor network environments. The techniques build on the observation that the values of the collected measurements exhibit similar patterns over time, or that different measurements are naturally correlated, as is the case between pressure and humidity in weather monitoring applications [16]. So, the data compression techniques lead to a reduction in the required inter-node communication, which is the main power consumer in WSNs [17].

Although the former algorithms conserve energy, they can lose much of the original structure in the data, providing only coarse statistics information. Furthermore, it is not necessary that users continuously extract all original data from the networks for analysis and decision-making. A prediction-based energy-efficient framework for data collection in clustered WSN was proposed. The practical algorithm for data aggregation avoids the need for rampant node-to-node propagation of aggregates, but rather it uses faster and more efficient cluster-to-cluster propagation [18]. The dimensionality reduction algorithm can project the monitoring data into a lower dimensionality representation while significantly decreasing the communication requirements [19–21]. In order to extract much more complete information about the shape and structure of sensor data than most aggregation schemes, the presentation model in sensor node can recognize local correlation in the measurements using kernel linear regression, where the support of a kernel function determines the set of monitoring data that are used to estimate basis function coefficient [22–24]. The coefficients of these basis functions and locations of kernels then provide a prediction for the behavior of the spatio-temporally correlated data common in WSN. However, a complete distributed linear regression-based hierarchical data gathering framework with fault-tolerant strategy and increment update scheme has never been proposed in detail. In this paper, relying on improved EADEEG protocol similar to LEACH (low-energy adaptive clustering hierarchy) [25], the CH node function was enhanced by integrating the linear regression model to minimize the communication cost. The data gathering framework not only implemented the nodes fault recognition by statistical hypothesis testing method when the mutation of the sensed data occurred, but also processed linear regression model update by relatively straightforward increment operation.

3 Network model and assumptions

The network model of the DLRDG framework was assumed that a set of N energy-constrained sensor nodes were randomly deployed in $M * M$ two-dimensional field. The following assumptions are made for the sensor network.

1. All sensor nodes are not mobile and unaware of their location.
2. The immobile sink node is only and considered to be a powerful node endowed with enhanced communication and computation capabilities and no energy constraints.
3. Communication from each node follows an isotropic propagation model.
4. Sensor nodes can adjust the transmitting power according to the distance, namely, radio transmitting power of nodes is controllable.
5. Sensor nodes can estimate the approximate distance by received signal strength.
6. Sensor nodes are fitted with the same radio communication model to simplify theory analysis. The radio channel is symmetric so that the energy required to transmit m-bit message from node i to node j is identical to the energy required to transmit m-bit message from j to i.
7. Sink node received the messages from the CH nodes using cluster-based hierarchical routing approach in WSN similar to LEACH. A subset of nodes is selected

CHs to facilitate communication functions. The multi-hop communication between CHs is different from the direct communication between CHs and sink like LEACH.

8. The communication radius of sensor nodes is more than a multiple of the cluster radius for implementing the direct communication between CH nodes.

In-cluster member nodes collect the monitoring messages and transmit the messages to their CH according to a reasonable nodes sleep scheduling strategy so the CH can compute the regression model coefficients using gathered history measurements and transmit the result to the sink node. When CH node received the monitoring data, it compared the error between the prediction data and actual measurement. If the error was paranormal, then CH node can activate fault-tolerant model to judge regression whether to update or not.

4 The proposed DLRDG framework

The existence of the spatial and temporal correlations brings significant potential advantages for the development and implement of efficient communication protocols well suited for the WSN paradigm. In this section, the data gathering framework is presented to exploit the spatio-temporal correlation characteristics of the clustered sensor network based on distributed linear regression technology that can model the approximation of the original data while significantly reducing the communication energy consumption.

4.1 Research motivation

Depending on the specific WSN application, the physical phenomenon information with spatio-temporal correlation may be a prediction model approximated by sensor source nodes in case of applications such as event monitoring. In order to obtain a more accurate estimate of physical processes, each node will generate an amount of sensor data. For example, the node produces a reading every 1 min or 60 readings an hour, then one node will generate 1,440 readings in a day. Possible to be thought of or believed, sensor-rich network system (the number of node is more than 100 nodes) will generate an enormous amount of sensor information for requiring the lifetime to more than 2 years. If the monitoring data were represented by using multi-dimensional value (e.g., humidity, temperature, illumination, pressure in weather monitoring application), a leap of sensor data would lead to increase the communication energy cost between sensor nodes and the sink node. Thus, it is disadvantageous for prolonging the lifetime of WSN. In order to reduce the sensor data, switching to a slower sampling rate (sensor nodes sample monitoring data every 30 min) would reduce the amounts of data. However, the slower sampling rate may cause system to miss high-frequency or emergency event. The maladjustment is especially prominent in forest fire monitoring and some similar security control applications in WSN. An appropriate approach to extracting part of the monitoring measurements is to build an approximation model of this data in the sensor network and transmit only the model coefficients. The model coefficients can provide a representation structure of the original monitoring measurements. At the same time, organizing a collection of sensor nodes into multi-hop clusters can decrease the number of transmitted messages to the sink and reduce the communication cost. The combination of predicting and clustering strategy can be well suited to building the energy-efficient data gathering framework of the spatio-temporally correlated data in common application.

4.2 Cluster formation

In order to take advantage of the existence of nodes of different abilities inside a WSN, data gather processing makes use of the hierarchical protocol based on clustered architecture. The classical LEACH offers no guarantee about the placement and number of CH nodes. The proposed framework carried out a more novel algorithm, namely EADEEG (an energy-aware data gathering protocol for WSNs) that can achieve a better performance in terms of lifetime by minimizing energy consumption for communications and balancing the energy load among all nodes [26]. For each node in network, the time t of the CH request message sent can be computed from equation $t = k \cdot T \cdot E_{\text{aver}}/E_{\text{resi}}$, where k is a float randomly uniform distributed in the interval [0.9, 1], T is a predefined duration time parameter of CH selection mechanism, E_{aver} denotes the average residual energy of all neighborhood nodes in the cluster radius of node V_i, and E_{resi}denotes the residual energy of node V_t. The main parameter of the competition for CH election is $E_{\text{aver}}/E_{\text{resi}}$. While there are advantages to use the EADEEG distributed cluster formation algorithm, it offers no guarantee about the monitoring blind spot brought or the disconnected CH nodes in some situations. However, using an auxiliary parameter deg (degree of the sensor node) to associate with $E_{\text{aver}}/E_{\text{resi}}$ for the competition for CH election may produce better clusters throughout the network. In our scheme, each node V_i computes the average residual energy E_{aver} of all neighborhood nodes V_k from Eq. (1).

$$E_{\text{aver}} = \frac{1}{\deg} \sum_{j=1}^{d} E_{kr} \tag{1}$$

where E_{kr} denotes the residual energy of the node V_k. Then, the time t of the CH request message sent can be computed from Eq. (2), where E_{ini} denotes the initial energy of the node. The CH competition phase can complete in time T. From 0 to $T/2$, a majority of CHs are certain. After the phase (from $T/2$ to T), a few of the remaining nodes become CH according to the large ratio of node residual energy to initial energy.

$$t = \begin{cases} k \cdot T \cdot \frac{E_{\mathrm{aver}}}{E_{\mathrm{resi}}} \cdot \frac{1}{\mathrm{deg}+1} & E_{\mathrm{resi}} > E_{\mathrm{aver}} \\ \frac{T}{2} + k \cdot \frac{T}{2} \cdot \frac{E_{\mathrm{resi}}}{E_{\mathrm{ini}}} & E_{\mathrm{resi}} < E_{\mathrm{aver}} \end{cases} \tag{2}$$

The clustering process can be described as follows:

Step 1: the neighbor discovery phase, each node broadcasts respective message with communication radius, receive the message from its neighbor nodes, initialize E_{aver}, deg, and compute the time t according to Eq. (2)
Step 2: CHs find phase, when current time is lesser than t, the nodes have received CH message from the neighbor node, the nodes abandon the CH competition, else broadcast CH message.
Step 3: nodes ascription phase, other nodes apart from CH nodes broadcast join message to the CH with max energy and update the parent information. The CH nodes receive join message, update the deg and other parameters

The clustering process is completed. With the virtual backbone in the sensor network, only CHs are concerned with data transportation, and other cluster member nodes are free to pursue their sensing tasks. This procedure can reduce the network energy consumption.

4.3 Distributed linear regression model

In DLRDG framework, the nodes organize themselves into local clusters, with one node acting as the CH. All non-CH nodes transmit their data to the CH, while the CH receives data from the active cluster members, performs linear regression model for original measurements, estimates the error between the measurements and the model approximation, and chooses whether transmitting the model coefficients to the remote sink node. In this section, the processing principle of the distributed linear regression model was derived.

Select the last m sensory measurements at regular intervals. Suppose that we are given a set of m data points $(t_1, y_1), (t_2, y_2), \ldots, (t_m, y_m)$, where t_i are sampling time points, $y_i (i \in [1, m])$ are actual measurements subject to errors. For these sensor data, we determined a function $Y(t)$ such that the approximation errors $\delta_i = Y(t_i) - y_i$ are very small for $i = 1, 2, \ldots, m$. The form of the function $Y(t)$ depends on the application at hand. Here, we assume that the function $Y(t)$ can be written as $Y(t) = \sum_{j=1}^{n} \lambda_j F_j(t)$, where the number of summands n and the basis functions F_j are chosen based on specific problem. When the outcome is numeric, linear regression is a feasible technique to implement the approximation of actual data. A common choice is $F_j(t) = t^{j-1}$, so the equation can be written as a polynomial of degree $n - 1$ in t point, which can be meant the equation $Y(t) = \lambda_1 + \lambda_2 t + \lambda_3 t^2 + \cdots + \lambda_n t^{n-1}$. By choosing $n = m$, we can calculate each y_i exactly. However, such high-degree $Y(t)$ fit the noise into the sensory data and generally gives poor results when used to predict y for previously unseen values of t. It is usually better to choose n significantly smaller than m and hope that by choosing the coefficients λ_i well, we can obtain the approximation values of measurements y_i. For example, we can fit a cubic polynomial to the last 60 measurements: $Y(t) = \lambda_1 + \lambda_2 t + \lambda_3 t^2 + \lambda_4 t^3$. Then, we only need to extract 4 parameters from the execution nodes: $\lambda_1, \lambda_2, \lambda_3$, and λ_4. Rather than transmitting 50 original measurements to sink node, the nodes of processing the function from the WSN communicate constraints on 4 parameters to further reduce the communication energy cost. We transform the polynomial model into the linear regression model using matrix representation. So the processing nodes do not calculate the high-degree polynomial and only need to execute the maintenance of the correlative matrix. Let $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)^T$ denotes the desired n vector of coefficients, $y = (y_1, y_2, \ldots, y_m)^T$ denote m vector of the actual measurements, the value matrix of the basis functions at the corresponding sampling time points was denoted as matrix $\boldsymbol{M}$:

$$\boldsymbol{M} = \begin{bmatrix} F_1(t_1) & F_2(t_1) & \cdots & F_n(t_1) \\ F_1(t_2) & F_2(t_2) & \cdots & F_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ F_1(t_m) & F_2(t_m) & \cdots & F_n(t_m) \end{bmatrix}$$

where the matrix elements $m_{ij} = F_j(t_i)$, let $\boldsymbol{Y} = (Y(t_1), Y(t_2), \ldots, Y(t_m))^T$denote the m vector of predicted values at t_i sampling time points, then

$$\boldsymbol{Y} = \begin{bmatrix} Y(t_1) \\ Y(t_2) \\ \vdots \\ Y(t_m) \end{bmatrix} = \boldsymbol{M}\lambda$$
$$= \begin{bmatrix} F_1(t_1) & F_2(t_1) & \cdots & F_n(t_1) \\ F_1(t_2) & F_2(t_2) & \cdots & F_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ F_1(t_m) & F_2(t_m) & \cdots & F_n(t_m) \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} \tag{3}$$

Thus, Eq. (4) is the m vector of approximation errors $\boldsymbol{\delta}$.

$$\boldsymbol{\delta} = \boldsymbol{M}\boldsymbol{\lambda} - \boldsymbol{y} \tag{4}$$

To minimize approximation errors, we choose to minimize the norm of the error vector $\boldsymbol{\delta}$. Thus

$\mathrm{Min}\left(\|\delta\| = \left(\sum_{i=1}^{m} \delta_i^2\right)^{1/2}\right)$, because $\mathrm{Min}\left(\|\delta\|^2 = \|\boldsymbol{M\lambda} - \boldsymbol{y}\|^2 = \sum_{i=1}^{m}\left(\sum_{j=1}^{n} m_{ij}\lambda_j - y_i\right)^2\right)$, we can minimize $\|\boldsymbol{\delta}\|$ by differentiating $\|\boldsymbol{\delta}\|^2$ with respect to each $\lambda_k (k = 1, 2, \ldots, n)$ and then setting the result to 0, namely, following Eq. (5):

$$\frac{d\|\delta\|^2}{d\lambda_k} = \sum_{i=1}^{m} 2\left(\sum_{j=1}^{n} m_{ij}\lambda_j - y_i\right) m_{ik} = 0,\ k = [1, n] \tag{5}$$

According to Eq. (4), Eq. (5) is equivalent to the single matrix equation $(\boldsymbol{M\lambda} - \boldsymbol{y})^T \boldsymbol{M} = 0$ or $\boldsymbol{M}^T(\boldsymbol{M\lambda} - \boldsymbol{y}) = 0$, namely,

$$\boldsymbol{M}^T\boldsymbol{M\lambda} = \boldsymbol{M}^T\boldsymbol{y} \tag{6}$$

Because the predefine basis function $F_j(t) = t^{j-1}$, the matrix $\boldsymbol{M}$ has full column rank, then $\boldsymbol{M}^T\boldsymbol{M}$ is positive definite as well. Namely, $(\boldsymbol{M}^T\boldsymbol{M})^{-1}$ exists, and the solution to Eq. (6) is Eq. (7).

$$\boldsymbol{\lambda} = (\boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T\boldsymbol{y} \tag{7}$$

Let

$$\boldsymbol{A} = \boldsymbol{M}^T\boldsymbol{M} = \begin{bmatrix} \langle F_1F_1\rangle & \langle F_1F_2\rangle & \cdots & \langle F_1F_n\rangle \\ \langle F_2F_1\rangle & \langle F_2F_2\rangle & \cdots & \langle F_2F_n\rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle F_nF_1\rangle & \langle F_nF_2\rangle & \cdots & \langle F_nF_n\rangle \end{bmatrix} \tag{8}$$

$$\boldsymbol{z} = \boldsymbol{M}^T\boldsymbol{y} = \begin{bmatrix} \langle F_1y\rangle \\ \langle F_2y\rangle \\ \vdots \\ \langle F_ny\rangle \end{bmatrix} \tag{9}$$

We can transform Eq. (7) to $\boldsymbol{\lambda} = \boldsymbol{A}^{-1}\boldsymbol{z}$, namely:

$$\boldsymbol{A\lambda} = \boldsymbol{z} \tag{10}$$

The term $F_j, j = 1, 2, \ldots n$ is the predefine basis function, the term $\boldsymbol{A}$ is the dot-product matrix, where each element denotes the dot product between two basis functions. The term $\boldsymbol{z}$ is the projected measurement vector, where each element is simply the projection of the measurement vector into the space of a particular basis function. Thus, given the measurement vector and the basis functions, we can calculate the coefficients of the optimal linear regression model $\boldsymbol{A\lambda} = \boldsymbol{z}$ with simple matrix operations.

4.4 Model update operations

For the event monitoring application in WSN, the volume of sensor data from the monitoring environment would become more and more excessive for the limited power supply and memory space of the sensor device. We may maintain the sampling data at a certain interval according to the processing ability of the sensor nodes. With linear regression model prediction, the CH nodes can selectively send its model parameters of estimating data distribution to sink node. Given an error bound $\varepsilon > 0$, CH node implements the fault-tolerant module to choose whether to update the regression model or reject the false sampling data if $|Y(t_i) - y_i| > \varepsilon$. The intuition of this choice is that if the sampling value is close to the predicted value there is no much benefit by transmitting it. If the monitoring sampling data are much different from the predicted value, it is necessary to update the linear regression model for recomputation of sensor data distribution. Very complex model update operation is not practical in data gathering framework of the WSN due to the limited computational capacity of sensor nodes. Fortunately, simple linear increment operation is sufficient to update the regression model by the matrix operations. Suppose that we have computed the dot-product matrix $\boldsymbol{A}$ of the basis functions and the projected measurement vector $\boldsymbol{z}$ at times $t_1, t_2, \cdots, t_{m-1}$, and we observe a new monitoring measurement at time t_m as following:

$$\boldsymbol{A}(t_m) = \begin{bmatrix} \langle F_1(t_m)F_1(t_m)\rangle & \cdots & \langle F_1(t_m)F_n(t_m)\rangle \\ \langle F_2(t_m)F_1(t_m)\rangle & \cdots & \langle F_2(t_m)F_n(t_m)\rangle \\ \vdots & \vdots & \vdots \\ \langle F_n(t_m)F_1(t_m)\rangle & \cdots & \langle F_n(t_m)F_n(t_m)\rangle \end{bmatrix}$$

$$\boldsymbol{z}(t_m) = \begin{bmatrix} \langle F_1(t_m)\boldsymbol{y}(t_m)\rangle \\ \langle F_2(t_m)\boldsymbol{y}(t_m)\rangle \\ \vdots \\ \langle F_n(t_m)\boldsymbol{y}(t_m)\rangle \end{bmatrix}$$

So the dot-product matrix $\boldsymbol{A}$ of the basis functions and the projected measurement vector $\boldsymbol{z}$ at times $t_1, t_2, \ldots, t_m$ are updated by the increment operation expression (11).

$$\boldsymbol{A} \leftarrow \boldsymbol{A} + \boldsymbol{A}(t_m), \quad \boldsymbol{z} \leftarrow \boldsymbol{z} + \boldsymbol{z}(t_m) \tag{11}$$

The scale of the linear regression model was controlled by using the time sliding window. That is, the coefficients of the basis functions were calculated respecting the measurements performed in the last T min. Similar to the operation expression (11), if measurement t_1 falls outside our time sliding window, the matrix $\boldsymbol{A}$ and vector $\boldsymbol{z}$ were updated according to the expression (12).

$$\boldsymbol{A} \leftarrow \boldsymbol{A} + \boldsymbol{A}(t_1), \quad \boldsymbol{z} \leftarrow \boldsymbol{z} + \boldsymbol{z}(t_1) \tag{12}$$

Thus, the CH node can extract the coefficients of linear regression model at any time by solving the linear system $\boldsymbol{A\lambda} = \boldsymbol{z}$ as well as update the dot-product matrix $\boldsymbol{A}$ of the basis functions and the projected measurement vector $\boldsymbol{z}$ by implementing the increment operations.

4.5 The fault-tolerant scheme

One of the key challenges in detecting event in a WSN is how to devise a strategy to handle the sudden or dramatic change of monitored sampling data. When the degree of correlation among neighboring sensor nodes varies spatially and the consecutive sensor readings of a particular sensor depict a smooth variation over time, then the sensor data present the spatio-temporally correlation. It is necessary to exploit spatio-temporal characteristics of sensor data to detect the emergence of event boundary accurately and quickly transmit the information to the sink node. In the DLRDG framework, CH node executed a fault-tolerant strategy based on statistical hypothesis testing for eliminating faulty readings or dealing with the regression model update. Using the stochastic process to describe the temporally correlation of the monitoring event, the fault sensor nodes in event region were identified by inter-comparison of the matching degree between local data reading sequence and event statistical characteristics.

The sensor reading approximations from the linear regression model were characterized as a sequence in sampling times $t_1 : \{Y(t_i)\}$, $i = 1, 2, 3, \ldots$. The expectation and variance of the sampling values sequence can present the statistical characteristic of the stochastic process. Suppose that $\varphi(x)$ is the probability density function of the random variable ξ. We define the function $\eta = f(\xi)$, thus, the mathematical expectation of the continuous random variables η can be defined as the equation

$$E\eta = Ef(\xi) = \int_{-\infty}^{+\infty} f(x)\varphi(x)\mathrm{d}x \qquad (13)$$

From Eq. (13), firstly, mathematical expectation $Ef(\xi)$ can be computed after the probability distributions of the random variables function $f(\xi)$ are derived. Considering the complexity of the above process, we can prove that the computing results using the distributions sum of the random variable ξ are the same as using Eq. (13). So the probability distributions of the function $f(\xi)$ need not be computed beforehand.

The expected value of a random variable does not characterize how "spread out" the variable's values are. We regard the sensor data as the random variable. The unconventionality sensor data can be identified by analyzing the departure between the sampling data and the mean. The notion of variance mathematically expresses how far from the mean (mathematical expectation) a random variable's values are likely to be. Similar to the expectation definition, the probability density function of the continuous random variable ξ is $\varphi(x)$, we define the function $\eta = f(\xi)$, and thus the variance can be defined as Eq. (14).

$$\mathrm{Var}(\eta) = \mathrm{Var}(f(\xi)) = \int_{-\infty}^{+\infty} [x - Ef(\xi)]^2 \varphi(x)\mathrm{d}x \qquad (14)$$

One of the key tasks in event monitoring of the WSN is how to detect whether the urgent event of interest is occurring. Therefore, the mathematical expectation function $E\mathrm{d}_{\mathrm{event}}(t_i)$ and the square root function $\mathrm{VarSq}_{\mathrm{event}}(t_i)$ of the variance in each normal sensor node can be obtained by the premeditated values from sink node. The occurrence threshold and the minimized continuance of the urgent event can be denoted by $T_{\mathrm{event}}(0)$ and $T_{\mathrm{conti}}(0)$, respectively. When a sequence $\{Y(t_i)\}$ of the sensor reading exceeded the occurrence threshold $T_{\mathrm{event}}(0)$ and the predefined count T_{count} which is the number of $\{Y(t_i)\}$ satisfying Eq. (15) condition, the sequence $\{Y(t_i)\}$ was considered as following the statistical hypothesis testing conditions.

$$\frac{|\mathrm{d}(t_i) - E\mathrm{d}_{\mathrm{event}}(t_i)|}{\mathrm{VarSq}_{\mathrm{event}}(t_i)} < \delta \qquad (15)$$

Equation (15) denoted the correlation degree between the sequence $\{Y(t_i)\}$ and the statistical characteristics of random event process. After the continuance T_{conti}, the CH nodes identified that the monitoring event was actually occurring and sends the urgent messages to sink node when the most cluster members have monitored the mutation of the sensed data. Subsequently, start the update model of linear regression model. The CH nodes maintain only $E\mathrm{d}_{\mathrm{event}}(t_i)$ and $\mathrm{VarSq}_{\mathrm{event}}(t_i)$ at $T_{\mathrm{conti}}/\Delta T$ integer ratio because of the limited memory space of sensor nodes.

The following two roles are, respectively, used to identify the misjudgment of the sensor node in the actual event region, and then the node was marked as the fault node. Firstly, a sequence $\{Y(t_i)\}$ of the sensor reading exceeded the occurrence threshold $T_{\mathrm{event}}(0)$, but it did not satisfy the statistical hypothesis testing conditions. Secondly, a sequence $\{Y(t_i)\}$ of the sensor reading satisfied the statistical hypothesis testing conditions, but the most neighbor nodes did not detect the mutation occurrence of sensor reading after the continuance T_{conti}. If the fault node is cluster member node, the CH does not receive the messages from it. If the fault node is the leaf CH node in the routing tree, it may delete directly. However, the parent CH node sends the rerouting message when the fault node is in the midst of the routing tree.

4.6 Data approximation process

The aim is to develop better data extraction and approximation algorithms for energy savings in sensor networks. These lead to lesser packet transmissions and reduce redundancy, thereby helping in increasing the network lifetime. The data approximation process is performed at cluster-based sensor network that CHs are selected for data extraction and transmission instead of other nodes in the network. As an example of presenting the data approximation process by linear regression model, suppose that we have six relative humidity sensor readings (the unit of relative humidity data is %) during the sampling time t_1–t_6, which (1, 17.72), (2, 21.71), (3, 18.88), (4, 18.58), (5, 22.34), (6, 19.79) shown as black circle in Fig. 1. In order to predigest computing, we wish to fit these relative humidity sampling points with a cubic polynomial $Y(t) = \lambda_1 + \lambda_2 t + \lambda_3 t^2 + \lambda_4 t^3$. Firstly, the basis functions matrix $\boldsymbol{M}$ is as follows:

$$\boldsymbol{M} = \begin{bmatrix} 1 & t_1 & t_1^2 & t_1^3 \\ 1 & t_2 & t_2^2 & t_2^3 \\ 1 & t_3 & t_3^2 & t_3^3 \\ 1 & t_4 & t_4^2 & t_4^3 \\ 1 & t_5 & t_5^2 & t_5^3 \\ 1 & t_6 & t_6^2 & t_6^3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 \\ 1 & 3 & 9 & 27 \\ 1 & 4 & 16 & 64 \\ 1 & 5 & 25 & 125 \\ 1 & 6 & 36 & 216 \end{bmatrix} \tag{16}$$

Secondly, the model coefficient vector was computed by Eq. (10) in Sect. 4.3, that is, $\lambda = (15.92, 3.2294, -0.80738, 0.066111)$. Therefore, the cubic polynomial is Eq. (17). The real line denotes the regression estimate curve of six relative humidity values in Fig. 1. The six approximations are: (1, 18.408), (2, 19.678), (3, 20.127), (4, 20.15), (5, 20.146), and (6, 20.51).

$$Y(t) = 15.92 + 3.2294t - 0.80738t^2 + 0.066111t^3 \tag{17}$$

Figure 2 shows that the absolute value of estimate error only with six measurements is relatively unsatisfied. Since the scale of the regression model is relatively less, the control about estimate error is not effective sensitivity. How large the regression model scale needs to be set according to the computing ability of sensor node and the confidence interval of the regression estimate. The CH node maintains the dot-product matrix $\boldsymbol{A}$ and projected measurement vector $\boldsymbol{z}$ of the basis function to implement the regression estimate. The data approximation process shown in Algorithm 1 is at the core of our framework.

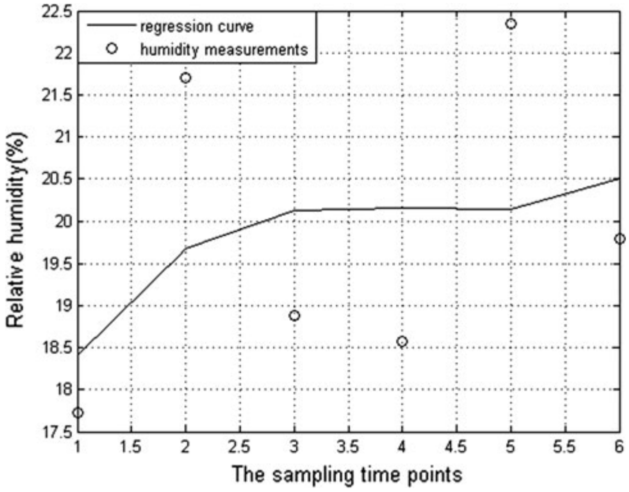

Fig. 1 The regression estimate of six relative humidity values

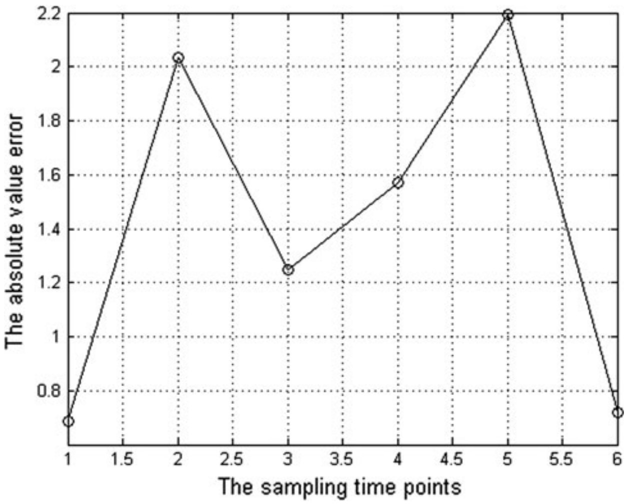

Fig. 2 The absolute value error of six relative humidity values

Algorithm 1: The linear regression model

```
Sensor_Distri_Regression(NodeElem i)
{//initialize parameters
Max_Time_W←size of the sampling time sliding window;
Message_itv←timer interval to send messages
Cluster_ID ← the cluster ID number of sensor nodes;
M ← the basis functions matrix;
For( each node i) do {A←0;    z←0; λ←0;}
For ( each node i) do //build the regression model
  { if (T<Max_Time_W)
  {A=A+A(T); z=z+z(T); }
  else { A=A-A(T1); z=z-z(T1); A=A+A(T); z=z+z(T);} T++;
For (each Message_itv)
{ $\lambda = A^{-1}z$;      Send_Message(CH_ID, λ);   }   }
For (each CH node j) do // the control about estimate error
{ if (Y(T) $\in \alpha$ ) Received_Message = true;
  Else
{ Received_Message = false;
  Send_alarm_Record(Sink, CH_ID, Node_ID);
Fault_tolerant();   }}}
```

Algorithm 2 shows the pseudocode description of the algorithm at the CH node. The CH maintains a set of history data for each cluster member and builds the linear regression model to predict measurement values. Furthermore, the CH node forwards the model parameters to the next CH node at routing tree. Algorithm 3 shows the pseudocode description of the algorithms at each cluster member. Each cluster member node senses the environment readings from the monitoring region according to the sleep scheduling strategy and sends the messages to the CH node. Algorithm 4 shows the pseudocode description of the algorithm at the sink node.

Algorithm 2: The algorithm at the CH node

```
for each CH node do
{
wait for receiving messages;
 if(message from BS)
 update processing parameters;
 else
     if(message from in_cluster node)
      {gather sampling data and predict data;
      sensor_Distri_Regression( );
      forwarding the model parameters to next CH; }
   Else
   farwarding data to next CH;
}
```

Algorithm 3:The algorithm at the cluster member

```
for each in_cluster node do
{
if(node_state=sleep)
 wait for wake_up messages;
 else
   if(energy_currenti=1)
   sendmessage(nodei,CHi);
 else
 energy_currenti=0;
}
```

Algorithm 4:The algorithm at the Sink node

```
for BS node do
{ while(1)
{wait for data from CHi;
if(received data ratio>threshold)
{send parameter to CHi;
continue;}
else
save received data; }}
```

4.7 The complexity analysis of the DLRDG strategy

The hard core algorithm of the proposed data gathering framework will be executed to solve the monitoring reading approximation through the linear regression model using the matrix operations, including the matrix addition, matrix subtraction, matrix multiplication, and matrix inversion. For forming dot-product matrix $\boldsymbol{A}$, the matrix multiplication operation obtained the speedups by Strassen algorithm, which decreases the running time complexity from $O(n^3)$ to $O(n^{2.81})$. To solve the linear equation $\boldsymbol{A\lambda} = z$, we could compute $\boldsymbol{A}^{-1}$ and then multiply z by $\boldsymbol{A}^{-1}$, yielding $\boldsymbol{\lambda} = \boldsymbol{A}^{-1}z$ in the ordinary way. However, the LUP decomposition approach for solving the linear equation is numerically stable and has further advantage of being faster in practice. The main idea of the LUP decomposition of the matrix $\boldsymbol{A}$ is to find three matrices L, U, and P such that PA = LU, where L is a unit lower-triangular matrix, U is an upper-triangular matrix, and P is a permutation matrix. It has been proved that every non-singular matrix $\boldsymbol{A}$ possesses such decomposition by the matrix characteristics. Computing the LUP decomposition for the matrix $\boldsymbol{A}$ has the advantage that we can more easily obtain the solutions of the linear system. Because of the matrix $\boldsymbol{A}$ is a symmetric positive-definite matrix, we can prove that matrix inversion is no harder than matrix multiplication relies on some properties of symmetric positive-definite matrix. That is, suppose two real $n \times n$ matrices were multiplied in time $T(n)$, where $T(n) = \Omega(n^2)$ and $T(n)$ satisfies two regularity conditions $T(n+k) = O(T(n))$ for any k in the range $0 \le k \le n$ and $T(n/2) \le C \times T(n)$ for some constant $C < 1/2$. Then, we can compute the inverse of any real non-singular $n \times n$ matrix in time $O(T(n))$.

In DLRDG framework, the complexity is also related to the scale of the regression model when the dot-product matrix $\boldsymbol{A}$ and the projected vector z were transmitted between CH nodes in routing tree for improving the estimate precision. Let S be the scale of the sampling data of the model in the sensor network. The messages between any two CH nodes are, in the worst, of size $S^2 + S$, that is, dot-product matrix $\boldsymbol{A}$and the projected vector z. For a sensor network with N nodes, the sum of all messages required to propagate the regression model information throughout the network is, in the worst case, $2N(S^2 + S)$. If we would like to upload the regression model coefficients to the sink node, we need, in the worst case, a total of $d \times n$ additional communication, where d is the depth of the routing tree, as the coefficient of each model has to be propagated to the sink node, and n is the regression model coefficient count.

5 Experiment results and performance evaluation

This section evaluates the validity and network energy consumption of the proposed DLRDG framework. To analyze the validity of the linear regression strategy, we implemented it in small WSN, which was deployed at gymnasium. The sensor nodes sampled the respective humidity measurements in the sunshine and a rainy day. To evaluate the network energy consumption and the fault-tolerant capability, we conducted a series of experiments using the network simulator developed under NS2. Relying on improved EADEEG protocol, the CH node function in the DLRDG framework was enhanced by integrating the linear regression model and compared with LEACH, EA-DEEG protocol for presenting the network energy saving results.

5.1 The validity analysis of the DLRDG strategy

In this experiment, we ran linear regression strategy on a dataset of the monitoring samples of light, temperature, pressure, and humidity collected from the gymnasium of six sensors. As an example, the humidity measurements were estimated by the linear regression model to prove the validity of the DLRDG strategy. The atmospheric water vapor affects the humidity sensor readings at the same time points of different day. The humidity monitoring measurements have high spatio-temporally correlation whether it rains or not. We believe that the spatio-temporally correlation properties of this humidity dataset will also be present in many other applications.

Choosing two groups humidity records of the same node in sensor network at the same the sampling time on a sunny day and a rainy day, we ran the linear regression algorithm, respectively, to measure the error of the model constructed. Table 1 presents the corresponding monitoring schedule of the 20 sampling time points. The computing results of the linear regression parameters were shown in Table 2, where λ_s is the model parameter on a sunny day and λ_r is the model parameter on a rainy day. Figure 3 illustrates the regression estimate values with linear model over the data from one sensor node over a sunny day and a rainy day. Intuitively, these measurements represent the correlation across time and space, and disperse around the regression curve, which denotes the approximation of the sensor readings. Figure 4 shows the absolute value errors between the actual monitoring data and regression approximation. Whether or not, the absolute value errors do not overstep the limited of error bound (as a general rule, it can be set 5 %), in this example, the maximal error is less than 2.5 %. Therefore, the linear regression model algorithm is the effective solution to the prediction estimate of the sensor readings with the spatio-temporally correlation properties. Then, rather than transmitting original monitoring data to next CH node or the sink node, the nodes communicate constraints on four regression model parameters.

Table 1 The corresponding monitoring schedule

Sampling points	A sunny day	A rainy day
1	8:00	8:00
2	8:10	8:10
3	8:20	8:20
…	…	…
20	11:10	11:10

Table 2 The linear regression parameters

Parameter subscript	λ_s	λ_r
1	26.344	49.053
2	0.47159	−0.32495
3	−0.042866	0.045912
4	0.0012293	−0.0013763

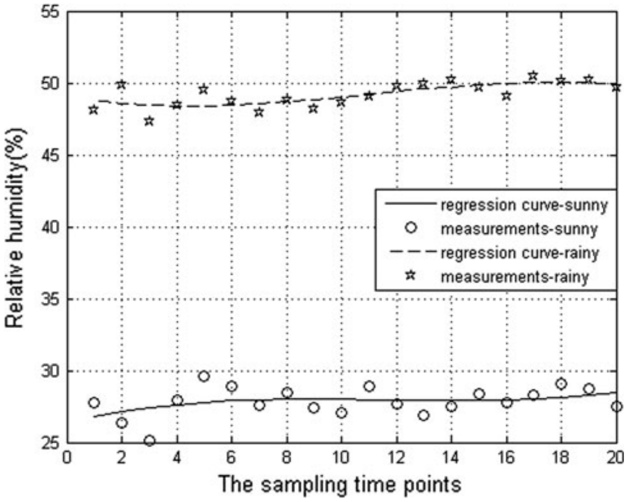

Fig. 3 The humidity regression curve of 20 sampling time points

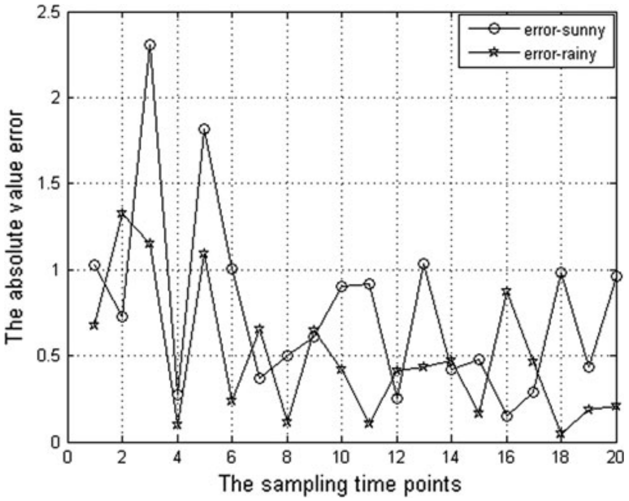

Fig. 4 The absolute value error between estimate values and measurements values

The estimate results of regression curves for different sampling time points with the same monitoring time segment on a sunny day were shown in Fig. 5. In order to formalize this effect of regression estimate, we select the root mean squared error (Eq. 18) to evaluate the precision of regression approximation. In addition to the sampling time points $m = 20$, the sample rate varies according to the different number of the sampling time points, such as $m = 10, 30, 40, 50$.

$$\omega = \sqrt{\left(\sum_{j=1}^{m} \left(Y(t_j) - y_{t_j}\right)^2\right) \Big/ m} \tag{18}$$

Figure 6 shows the root mean squared errors corresponding to the varying sampling time points. Notice that the linear regression model performs much better during the sampling time when the number of the sampling points is 30, the root mean squared error is minimum. In other words, for the estimate precision of regression model, the more the number of sampling points during the monitoring period, the error is not less. In general, lowering the sampling rate may cause the sensor network system to miss high-frequency events, switching to a higher sampling rate would increase the energy consumption and the sensitivity of the monitoring information. Thus, the number of the sampling points was chosen according to the different monitoring system requirement for the trade-off between the accuracy of the regression model and the energy cost.

5.2 The evaluation of the network energy consumption and fault-tolerant capability

In this subsection, we provide a particular analysis of the proposed DLRDG framework for the energy consumption. We compare the distributed linear regression-based strategy against the standard clustered LEACH and EADEEG protocol. Four main aspects of DLRDG framework were evaluated: the total energy dissipation of the CH nodes each round, the total energy dissipation of the CH nodes with different regression period, the total energy dissipation, and the fault-tolerant capability, which were analyzed separately. In each experiment, the NS-2 tool was used to implement and simulate the network system. We assume a simple model for the radio hardware energy dissipation where the transmitter dissipates energy to run the radio electronics and the power amplifier and the receiver dissipates energy to run the radio electronics [21]. Suppose the distance between the transmitter and receiver is d, according to the theory of wireless communication, if the distance d is less than a threshold d_{Thres}, the free space channel model (FS) is used (d^2 power less); otherwise, the multi-path model (MP) is used (d^4 power less). Thus, to transmit and receive h-bit message at d distance, the radio energy dissipation $E_{\text{TX}}(h, d)$ and $E_{\text{RX}}(h)$ were calculated, respectively, by Eqs. (19) and (20).

$$\begin{aligned} E_{\text{TX}}(h,d) &= E_{\text{TX-elec}}(h) + E_{\text{TX-am}}(h,d) \\ &= \begin{cases} h \times E_{\text{elec}} + h \times \varepsilon_s \times d^2 & d < d_{\text{Thres}} \\ h \times E_{\text{elec}} + h \times \varepsilon_m \times d^4 & d \geq d_{\text{Thres}} \end{cases} \end{aligned} \tag{19}$$

$$E_{\text{RX}}(h) = E_{\text{RX-elec}}(h) = h \times E_{\text{elec}} \tag{20}$$

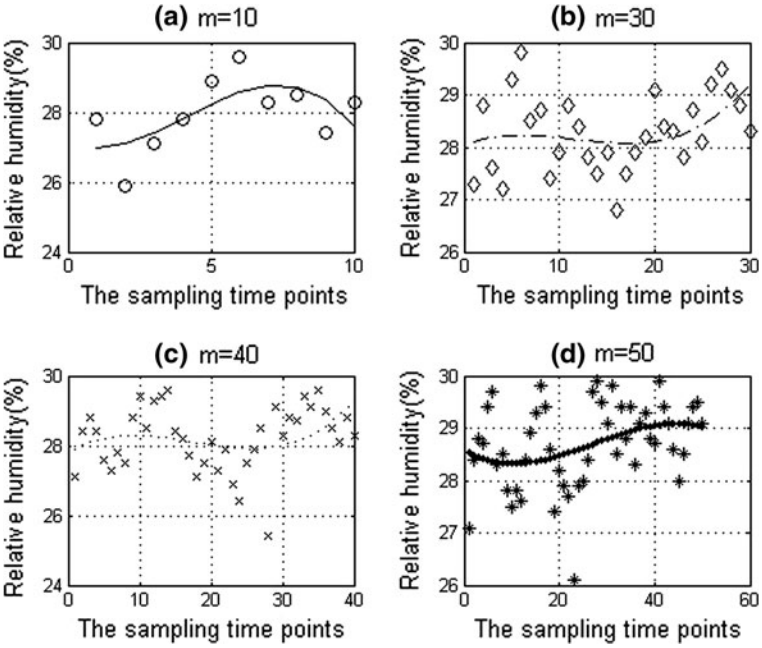

Fig. 5 The humidity regression curve at the varied sampling time points

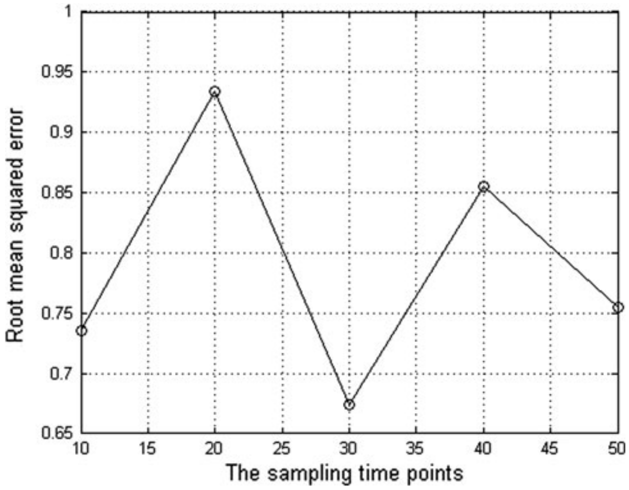

Fig. 6 The root mean squared error corresponding to the varied sampling time points

The electronics energy E_{elec} depends on factors such as the digital coding, filtering, modulation, and spreading of the signal, whereas the amplifier energy $\varepsilon_s \times d^2$ or $\varepsilon_m \times d^4$ depends on the distance to the receiver and the acceptable bit-error rate. In addition, to compute a regression estimate each round, the energy cost of the CH nodes operation is $E_{\mathrm{re}} = N_{\mathrm{ch}} \times E_{\mathrm{com}}$, where the number of CH nodes in network is N_{ch} and E_{com} is the energy cost of each CH node for computing the regression model. For the experiments described in this paper, the main parameters of the WSN simulation system are set as Table 3.

For these experiments, each node begins with only 2 J initial energy and 500 bytes messages to send to the sink node. The CH status was determined at the beginning of each round, which lasts for 20 s. Node energy is consumed whenever a sensor in network transmits or receives data or performs linear regression estimate model. Figure 7 shows how the total energy dissipation of the CH nodes each round varies as the simulation time runs on for the three data gathering network protocols (LEACH, EADEEG, and DLRDG). Obviously, the CH nodes of DLRDG framework required less energy in the simulation time than other two protocols. This is because a much smaller amount of data was transmitted to the sink by CH using the linear regression estimate model to provide a structured prediction of the measurements. The energy dissipation increased slightly in each regression period for computing the model coefficients. After the simulation time is 380 s, using LEACH protocol the energy dissipation of CH is rapid decline. The reason for this is that the CH node has touched the energy threshold during the current round, which did not performed transmit–receive operation and calculation. The graph shows the total energy dissipation of all CH nodes before the CH exhausted the initial energy.

Table 3 The main experiments parameters

Parameter	Acronym	Value
The sensor network region	$R \times R$	100 m × 100 m
The number of sensor node	N	100
Initial node energy	E_{ini}	2 J
The sink node location	Sink	(50,80)
The number of CH node	N_{ch}	5
TX/RX electronics constant	E_{elec}	50 nJ/bit
FS model amplifier constant	ε_s	10 pJ/bit/m^2
MP model amplifier constant	ε_m	0.0013 pJ/bit/m^4
Regression estimate energy cost	E_{com}	5 nJ/bit
The bandwidth of the channel	Bandwidth	1 Mb/s
Data message size	Data_size	500 bytes
Transmission delay	Tran_delay	25 µs
Simulation time	Sim_time	600 s
The scale of regression model	m	20
Regression period	R_period	60 s
The interval of each round CH	Round_time	20 s
CH energy threshold	CH_E_{thre}	10^{-4} J

When the error between the monitoring reading and prediction value exceeds the predefined error bound, the CH node must perform the fault-tolerant model for choosing whether to update regression model or not. How will the high-frequency model update operation impact on the energy dissipation of the CH node? For the sake of comparison with these schemes, the total energy dissipation of the CH nodes was evaluated while varying the regression period (that is 10, 30, and 60 s). Figure 8 shows that the total energy dissipation of the CH node is least when the regression period is 60 s. It stands to reason that if the regression period is to last less time the updated regression coefficients increase the communication cost of the CH node. Therefore, the DLRDG data gathering

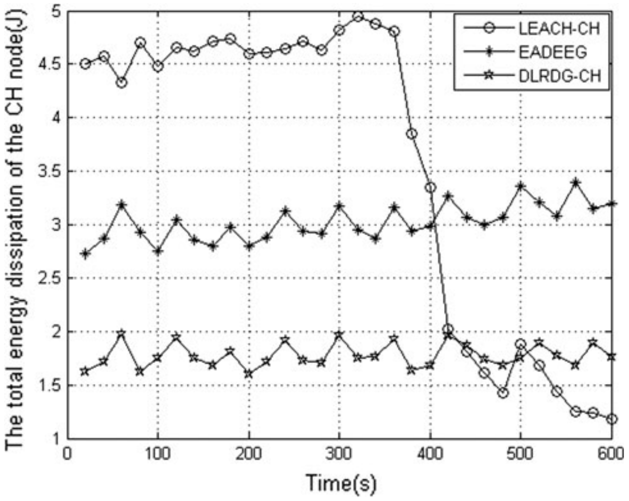

Fig. 7 The total energy dissipation of the CH node

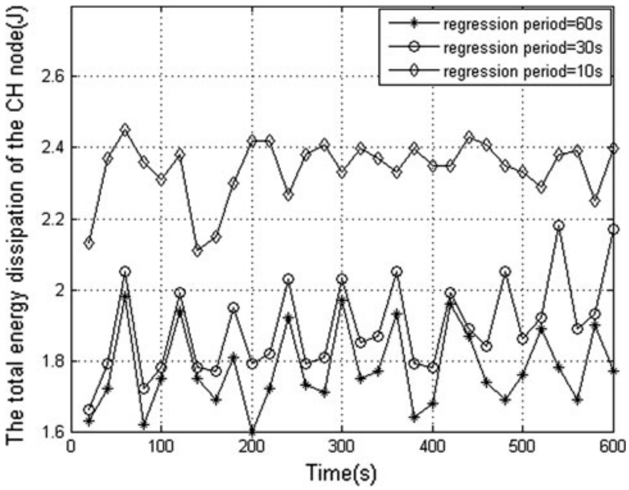

Fig. 8 The total energy dissipation of the CH with the varying regression period

framework is more efficient when the measurements present a linear variety in most cases and saves much more communication energy than other non-regression prediction schemes.

Figure 9 plots the total energy dissipation before sensor nodes exhaust the initial energy for DLRDG scheme, LEACH, and EADEEG, respectively. Our simulation results demonstrate that the DLRDG can achieve much more energy saving (about 100 J) in the lifetime of WSN when compared with other strategies. The energy dissipation of distributed regression grows quite slowly with the simulation time, unlike other two schemes. Note also that three experiments show an upward tendency, the total energy dissipation of LEACH protocol is rather slow increase during the simulation time terminal stage (after 380 s). The reason is the same with the commentary in the Fig. 7 due to the death of sensor nodes. Furthermore, we conducted an additional experiment to evaluate the real energy dissipation in the limitless node energy.

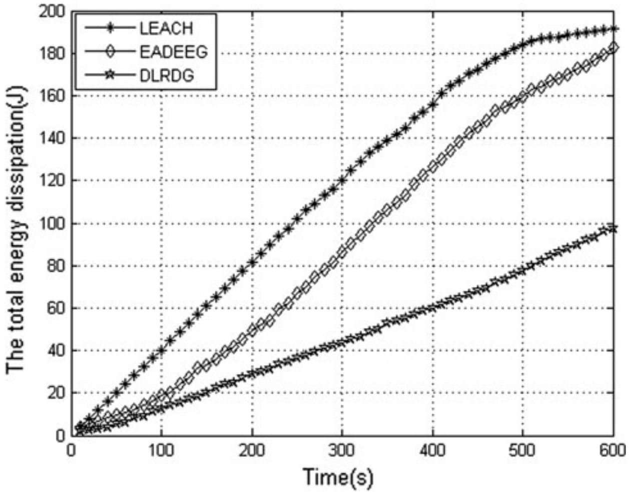

Fig. 9 The total energy dissipation with different schemes

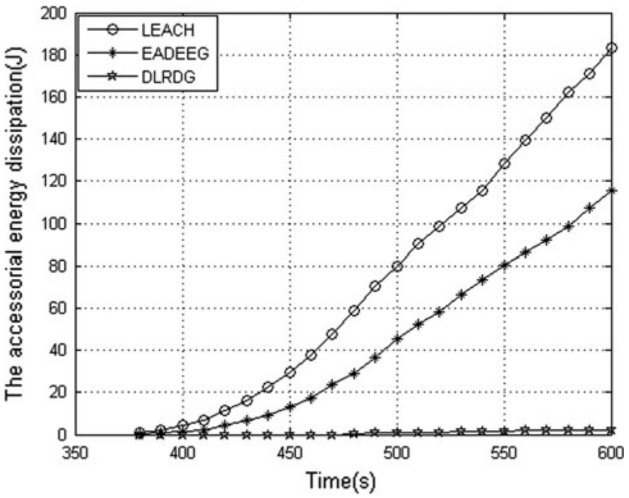

Fig. 10 The accessorial energy dissipation with different schemes

As shown in Fig. 10, the accessional energy dissipation of the DLRDG network remains little or nothing over the simulation time. While the accessional energy dissipations are about 180 and 120 J using LEACH and EADEEG, respectively. Hence, for most cases, the advantages of using DLRDG at energy consumption become more efficient when the sensor readings of depending on the application specific are linear variety.

However, the sensor nodes may have hardware failure or detect the occurrences of some critical event in the network domain. The mutation of sensor readings will result in fall short of the linear characteristic for regression model. An important challenge in the application of the DLRDG framework for a variety of environments monitoring is to identify the occurrence of faulty sensors and preventing the

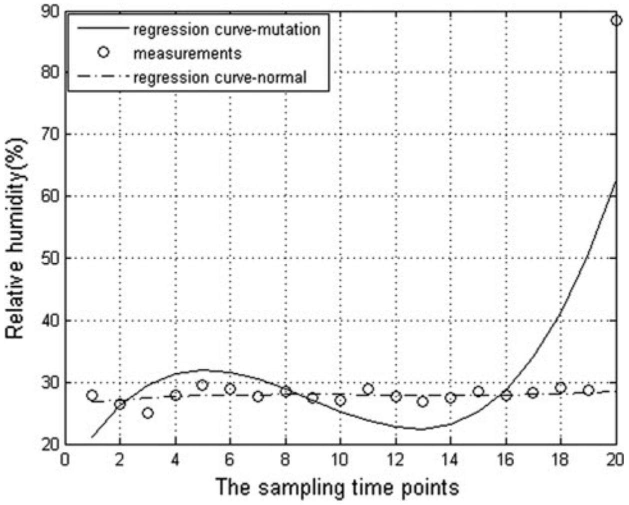

Fig. 11 The regression curve comparing between mutation and normal sampling

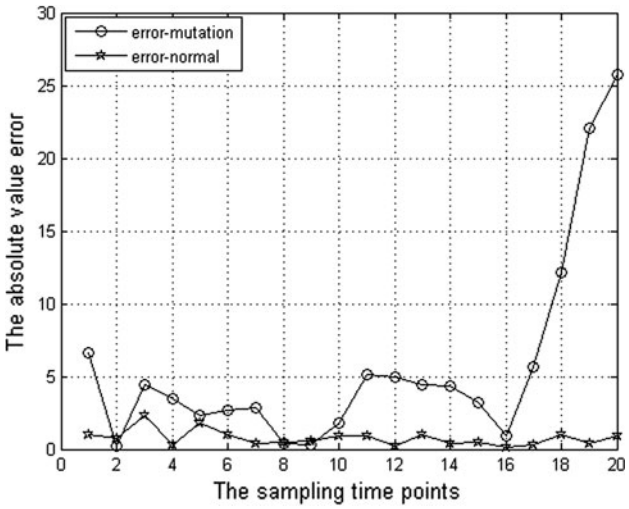

Fig. 12 The absolute value error comparing between mutation and normal sampling

fault readings to propagate further, transmitting the warning messages of the mutation readings to the sink node under control. Suppose there was a sudden change in the humidity reading from 28.7 to 88.5 %. Figure 11 shows that the regression estimate curve deviates from the actual measurements spot due to occurrence of the mutation reading.

The absolute value error between the measurements and regression estimate values with the normal sampling and the mutation sampling are shown in Fig. 12. For the normal sampling the variations in the absolute value error per round over time are small. On the contrary, for the mutation sampling, the absolute value error of the last sampling time point almost increases to 26 %, which has overstepped the certain prespecified error threshold.

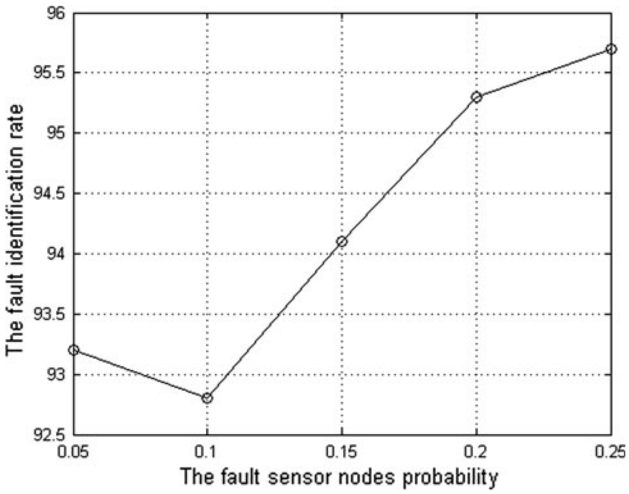

Fig. 13 The fault identification rate with the varying fault sensor nodes probability

The fault-tolerant model in the DLRDG framework was implemented to choose the next task of the CH node, which is to update regression for retransmitting coefficients or abandon the faulty reading. As stated before, the fault identification rate of the fault-tolerant model can have some effect on the result obtained about the prediction precision. So we also measured the fault identification rate of the DLRDG framework with varying fault probability of sensor nodes in network. Figure 13 shows the simulation results for the different settings where the fault sensor nodes probability from 5 to 25 %. In all cases, the fault identification rate keeps up well (about 95 %) in spite of the increase in the number of fault sensor nodes. The results of the further experiments demonstrated the correctness of the data gathering strategy in fault-tolerant model.

In general, using the DLRDG framework, we can achieve the satisfactory approximation at lower communication rate than suggested by the worst-case complexity of the framework.

6 Conclusions

In this paper, we have proposed and described a novel DLRDG framework for clustering-based data gathering. The strategy is relatively simple and general. The distributed linear regression model is used to implement the subtle trade-off between communication and calculation cost. Rather than delivering sensor samples at a continuous rate, as most application systems proposed, our scheme allows CH nodes to locally estimate the measurements that are very near to the prediction value by the linear regression model. After distributed regression computing, CH node has the coefficients of the estimate model to predict the approximation of the monitoring event. By less communication energy consumption, the network system can provide queries about the distant past or future by storing regression model coefficients, which determine a compact summary of sensor readings at a given point in time. For the sensor readings with linear character in WSN, a sample polynomial model is sufficient to represent the monitoring data. Experimental results indicate that the DLRDG framework obtained more than 60 % savings in the energy as compared with LEACH and EADEEG. In addition, on an average, 94 % of the total number of faulty sensors is detected right at the framework when the mutation reading occurs, thus preventing the fault sensor readings to be incorporated in the calculated regression model. To summarize, the DLRDG is a viable and energy-efficient framework to facilitate sensor monitoring data collection in clustering-based WSN.

There are several future research works. First, we plan to design a regression model with an adaptive control in the number of the sampling point according to the varying environment information for decreasing the model update operation rate. Second, we are seeking the more efficient algorithms to reduce the computation overhead of regression estimate. Third, we plan to integrate the optimization techniques to improve the quality of clustering and routing. Finally, we are interested in exploring this regression strategy (using non-linear regression scheme) in multimedia information represent of camera-based WSN. If these research contributions can be obtained, it is very important for the energy-efficient hierarchical data gathering techniques, which is a fruitful research area for the WSNs.

Acknowledgments The research work was supported by the National Natural Science Foundation of China under Grant No. 61070162 and 71071028, and open research fund of Key Laboratory of Complex System and Intelligence Science, Institute of Automation, Chinese Academy of Sciences under grant No. 20100106.

References

1. Wander AS, Gura N, Eberle H et al (2005) Energy analysis of public-key cryptography for wireless sensor networks. In: Proceedings of the third IEEE international conference on computing and communications, Seattle, pp 324–328
2. Estrin D (2005) Wireless sensor networks tutorial part IV: sensor network protocols. Invited speech of International Conference on Mobile Computing and Networking (Mobicom), Atlanta
3. Anastasi G, Marco C, Di Francesco M, Passarella A (2009) Energy conservation in wireless sensor network: a survey. Ad Hoc Netw 7(3):537–568
4. Xu XH, Li XY, Mao XF et al (2011) A delay-efficient algorithm for data aggregation in multihop wireless sensor networks. IEEE Trans Parallel Distrib Syst 22(1):163–175
5. Wu YW, Li XY, Liu YH et al (2010) Energy-efficient wake-up scheduling for data collection and aggregation. IEEE Trans Parallel Distrib Syst 21(2):275–287
6. Liu B, Ren FY, Lin C, Jiang X (2008) Performance analysis of sleep scheduling schemes in sensor networks using stochastic petri net. In: Proceedings of IEEE international conference on communications. Beijing, pp 4278–4283
7. Monaco U, Cuomo F, Melodia T et al (2006) Understanding optimal data gathering in the energy and latency domains of a wireless sensor network. Comput Netw 50(18):3564–3584
8. Bista R, Kim YK, Chang JW (2009) A new approach for energy-balanced data aggregation in wireless sensor networks. In: Proceedings of ninth IEEE international conference on computer and information technology, Xiamen, vol 2, pp 9–15
9. Ren HL, Meng MMQH (2006) Biologically inspired approaches for wireless sensor networks. In: Proceedings of IEEE international conference on mechatronics and automation, Luoyang, pp 762–768
10. Saleem M, Di Caro GA, Farooq M (2010) Swarm intelligence based routing protocol for wireless sensor networks: survey and future directions. Inf Sci 181(20):4597–4624
11. Al-Karaki JN, Ul-Mustafa R, Kamal AE (2009) Data aggregation and routing in wireless sensor networks: optimal and heuristic algorithms. Comput Netw 53(7):945–960
12. Zheng J, Wang P, Li C (2010) Distributed data aggregation using slepian-wolf coding in cluster-based wireless sensor networks. IEEE Trans Veh Technol 59(5):2564–2574
13. Konstantopoulos C, Mpitziopoulos A, Gavalas D et al (2010) Effective determination of mobile agent itineraries for data aggregation on sensor networks. IEEE Trans Knowl Data Eng 22(12):1679–1693
14. Lin K, Chen M, Zeadally S, Rodrigues JJPC (2012) Balancing energy consumption with mobile agents in wireless sensor networks. Future Gener Comput Syst 28(2):446–456
15. Jiang HB, Jin SD, Wang CG (2010) Parameter-Based data aggregation for statistical information extraction in wireless sensor networks. IEEE Trans Veh Technol 59(8):3992–4001
16. Deligiannakis A, Kotidis Y, Roussopoulos N (2007) Dissemination of compressed historical information in sensor networks. VLDB J 16(4):439–461
17. Srisooksai T, Keamarungsi K, Lamsrichan P, Araki K (2011) Practical data compression in wireless sensor networks: a survey. J Netw Comput Appl 35(1):37–59
18. Jiang HB, Jin SD, Wang CG (2011) Prediction or not? an energy-efficient framework for clustering-based data collection in wireless sensor networks. IEEE Trans Parallel Distrib Syst 22(6):1064–1071
19. Zhu H, Schizas ID, Giannakis GB (2009) Power-efficient dimensionality reduction for distributed channel-aware kalman tracking using WSNs. IEEE Trans Signal Process 57(8):3193–3207
20. Zhang HG, Quan YB (2001) Modeling, identification, and control of a class of nonlinear systems. IEEE Trans Fuzzy Syst 9(2):349–354
21. Zheng HP, Kulkarni SR, Poor HV (2011) Attribute-distributed learning: models, limits, and algorithms. IEEE Trans Signal Process 59(1):386–398
22. Guestrin C, Bodik P, Thibaux R et al (2004) Distributed regression: an efficient framework for modeling sensor network data. In: Proceedings of Third International Symposium on Information Processing in Sensor Networks, California, USA, pp 1-10
23. Mateos G, Bazerque JA, Giannakis GB (2010) Distributed sparse linear regression. IEEE Trans Signal Process 58(10):5262–5276
24. Zhang HG, Liu JH, Ma DZ, Wang ZS (2011) Data-core-based fuzzy min–max neural network for pattern classification. IEEE Trans Neural Netw 22(12):2339–2352
25. Heinzelman WB, Chandrakasan AP, Balakrishnan H (2002) An application-specific protocol architecture for wireless microsensor networks. IEEE Trans Wireless Commun 1(4):660–670
26. Liu M, Cao JN et al (2007) EADEEG: an energy-aware data gathering protocol for wireless sensor networks. J Softw 18(5):1092–1109 (in Chinese)