Ad Hoc & Sensor Wireless Networks, Vol.11(1-2), pp. 111–135
Reprints available directly from the publisher
Photocopying permitted by license only

Energy-efficient Lossy Data Aggregation in Wireless Sensor Networks*

JIANHUI ZHANG[1], XINGFA SHEN[1], GUOJUN DAI[1],
YUNXIA FENG[1], SHAOJIE TANG[2], CHANGPING LV[1]

[1]*Institute of Computer Application Technology, College of Computer and Software, Hangzhou Dianzi University, P.R.China.*
Email: jhzhang@ieee.org

[2]*Department of Computer Science, Illinois Institute of Technology, Chicago, IL60616, USA,*
Email: stang7@iit.edu

Received: November 18, 2009. Accepted: February 22, 2010.

In wireless sensor networks (WSNs), in-network data aggregation is an efficient way to reduce energy consumption. However, most of the existing data aggregation scheduling methods try to aggregate data from all the nodes in each time-instance, which is neither energy efficient nor practical because of the link unreliability and spatial and temporal data correlation. In this paper, we propose a new scheme allowing the data aggregation with the data loss. In our scheme, we selectively let some nodes sample and aggregate data, then transmit it to the sink. Two different cases are studied. Firstly, this paper assumes that the links are reliable and the error between the data of all nodes and that of sampled nodes is bounded. The detailed analysis is given on the error bound when the confidence level is given in advance. Secondly, this paper assumes that the links are unreliable with a certain probability. Then we obtain that the error is still bounded under a given confidence level when the probability of link unreliability is not too high or the success probability of retransmission is high enough. We also study how to assign the confidence level among the parent nodes such that each parent node can calculate the minimum number of sampling leaf nodes based on the corresponding confidence level. Through analyzing, we show that it can surely save energy to adopt our method when the link is reliable. When the link is not reliable, the energy still can be saved if the success probability of

*The conference version of this paper was previously accepted by MSN'09.

retransmission is high enough. The performance evaluation by simulation is discussed in the end of this paper.The results of the simulation indicate that it can save energy and does not effect the data accuracy to adopt our scheme if a certain bounded error is acceptable. Since the data redundancy often happens in WSNs, it is feasible to allow certain data error.

Keywords: Lossy Data Aggregation; Energy Efficiency; Data Sampling; Data Loss; Wireless Sensor Networks

I INTRODUCTION

Wireless sensor networks (WSNs) are resource constrained: limited energy, bandwidth, memory and so on. The energy consumption caused by the data processing by a sensor node is usually much less than that caused by the communication [1]. Thus, it is a common way to save the energy consumption by reducing the communication. An effective method is to apply data aggregation [2] or compression before transmitting data. A number of novel data aggregation methods have been proposed recently with various optimization goals, such as reducing the energy consumption [2, 3], and the delay of data aggregation [4, 5].

Most of data aggregation schemes assume that all the sampled data can be successfully transmitted to the sink [6]. In practice, wireless links are not always reliable and not all of the nodes can work normally all through. Furthermore, the sample rate is always limited by the bandwidth. Therefore, some packets are inevitably lost at some unpredictable time slots on some links. When a parent node collects data from others, some or all sensed data of a node may miss at some time slots. Unfortunately the final data received by the parent node is fragmentary. Furthermore, typical data aggregation schemes would let the parent nodes sample data at full time slots and collect the data from their children.

Although there are lots of work focusing on data aggregation [6, 7], a few of them synchronously consider the packet loss and sampling data loss, which are ineluctable in WSNs. Meanwhile the energy cost on data sampling and transmitting can be reduced when proper strategies are designed.

Another unavoidable question is whether a much precise information obtained from the practical surrounding by WSNs is worthy of the exiguous recourse in WSNs when the less precise information is acceptable. In many applications, the data sampling is better if the sampling period is longer. The longer sample period and lower duty ratio can prolong the network life time and save energy since it is important to save energy when the batteries of the nodes are hard to be recharged. In this sense, it is advisable to collect the data of a part of nodes at a part of periods.

In this paper, we introduce a new method into the data aggregation. It need only collect the information of a part of nodes and a part of time slots therefore it reduces the sampling time and energy of sampling and transmitting data.

In our method, we decompose the data aggregation tree into two kinds of basic components: *serial connection (SC)* and *collateral connection (CC)*. The data aggregation tree is constructed by selecting a connected dominating set (CDS) [8]. We divide the whole network life into a series of length-equivalent periods, each of which contains several time slots. Whether a sensor node samples data from environment in the time slots or not is decided by our data aggregation method. Based on the two basic components, we respectively analyze the necessary number of nodes or time slots to sample data when the error $\wp$ between the value D containing the sensed information and the corresponding value D' of the real world is given. we design algorithms to insure that each parent node in a data aggregation tree or subtree can decide the number of leaf nodes who need not sample data and that each node can know the number of time slots in which it need not sample data. Our algorithms guarantee that the error $\wp$ is less than an expected value E' with the least probability $1-\gamma$.

The rest of the paper is organized as follows. Section II outlines the relative work of the data aggregation and CDS. Section III presents the assumption and formulates the problem of loss data aggregation in the network. Our lossy data aggregation scheme is presented and analyzed in Section IV. We also give the simulation to evaluate the performance of our schemes in Section V. Section VI concludes the whole paper.

II BACKGROUND

A Data Aggregation

In WSNs, data aggregation has been well studied in recent years [6], [4,9–11]. A main purpose of in-network aggregation is to decrease the transmission power consumption [2]. Instead of transmitting raw data to sink, it can save much energy and decrease network interference to compute and transmit partially aggregated data in network. [12] proposed a heuristic algorithm for constructing data aggregation trees that minimize total energy cost under the latency bound and compute the worst case delay for a sensor node to aggregate the data from all its child nodes in the aggregation tree based on an analytic model for IEEE 802.15.4 standard. In order to decrease the time latency more, [5] developed a distributed collision-free schedule with the latency bound of $24D+6\Delta+16$, where D is the network diameter and Δ is the maximum node degree among the network. The tradeoff between energy consumption and time latency was considered in [13]. To balance the tradeoff, [14] imposed a decision-making problem in aggregation, and proposed a semi-Markov decision process model to analyze the decision problem and determine the optimal policies at nodes with local information.

Because the low bandwidth and energy limitations are inherent to sensor networks, an adaptive application-independent data aggregation (AIDA)

component, fitted into the sensor network communication stack, is developed in [15] to maximize the utilization of the channel while the energy is saved. In fact, the channel states are dramatically affected by the radio states (transmitting, receiving, listening, sleeping and being idle) of a transmitter and the environment. Based on TDMA MAC layer protocol, Wu, Li *etc* scheduled the sensor nodes at different radio states [16]. The energy consumed by their scheduling for homogeneous network is at most twice of the optimum and for heterogeneous network is at most $\Theta(\log \frac{R_{max}}{R_{min}})$ times of the optimum. They also proposed data gathering scheme to grantee the energy consumption and the network throughput within a constant factor of the optimum. However, it is costly to efficiently use TDMA model in WSNs since it consumes much resource to implement synchronization protocol in the network. So the collision and interference are unavoidable in WSNs [17]. [18] designed a collision-free scheduling when data collection was implemented.

B CDS

The CDS problem has been widely studied in Unit Disk Graphs (UDG) [19]. Before constructing CDS, many existing algorithms firstly found a Maximal Independent Set (MIS) I based on a given network and then connected all nodes in I to form a CDS [20]. In fact, the communication ranges of different sensor nodes differ from each other not like UDG, which results in the symmetric communication links in multihop wireless networks. [19] presented two algorithms having constant performance ratios for its size and diameter of the constructed CDS. [20] solved the link asymmetric problem by constructing a strongly CDS (SCDS) and presented a polynomial-time $(3H(n-1)-1)$-approximation algorithm for minimum SCDS, where H is the harmonic function. In order to save energy, [21] introduced a notion of an extended dominating set (EDS) where each node in an ad hoc network is covered by either a dominating neighbor or several 2-hop dominating neighbors. It also gave the heuristic solutions to the ECDS/EWCDS based on Guha and Khuller's MCDS.

In ad hoc or sensor network, the node and edge are weighted, such as different edges have different traffic or consume different energy on communication, which results in different interference range. [22] presented a polynomial-time algorithm approximating the minimum weight edge dominating set problem within a factor of 2. Instead of minimizing the backbone size, [7] proposed an efficient distributed method to construct a weighted backbone with low cost. The total cost of the constructed backbone is within a small constant factor of the optimum for homogeneous networks. The total number of messages of our method is $O(n)$ when the geometry information of each wireless node is known and the total number of messages is $O(m)$ otherwise for a network of n devices and m communication links.

III PRELIMINARY

A Network Model

We consider a network consisting of n sensor nodes, which are randomly and uniformly deployed in a $\mathcal{C} \times \mathcal{C}$ area. We denote a node as N_i, $i = 1, \cdots, n$, which compose a node set $V = \{N_i | i = 1, \cdots, n\}$. Every node has a transmission range r such that two nodes N_i and N_j can communicate directly if $||N_i - N_j|| \leq r$ and there is no interference. The transmission range r of each node is properly set to guarantee the network connected [23]. We construct a data aggregation tree by adopting an existing algorithm of selecting a CDS $\mathcal{S}$, $\mathcal{S} \subset V$. Wan *etc* presented a distributed algorithm that has an approximation factor of at most 8, $O(n)$ time complexity and $O(nlogn)$ message complexity [24]. Then each node N_i ($N_i \notin \mathcal{S}$) can find a node N_j ($N_j \in \mathcal{S}$) and connect with it. All nodes in $N_j \in \mathcal{S}$ can connect together and send their aggregated data to the sink by multihop fashion. We call the nodes in $\mathcal{S}$ as parent nodes and the nodes not in $\mathcal{S}$ as leaf nodes. All nodes send their data to their parent nodes and the parents send their data the sink by relay nodes. Since those nodes not in $\mathcal{S}$ connect to those in $\mathcal{S}$, the sink has no node without leaves.

B Data Aggregation Scheme

In WSNs, the primary task is to collect and transmit data to the destination. Meanwhile the network often works in the duty cycle style in order to prolong the network lifetime as shown in Figure 1. A node keeps itself "active" in work time t_w and "sleep" in sleep time t_s. Furthermore, the work time t_w is equally divided into K time slots. All nodes run the same duty cycles and sample data in the synchronization way.

We denote the life time of the network works by L_T and assume that $L_T = L \times T$, where L is a positive natural number and T is the period. At p^{th} ($p = 1, \cdots, K$) time slot S_p of q^{th} ($q = 1, \cdots, L$) period T_q, a node N_i samples a data from the surrounding. The data is denoted as $D(p, q, i)$ so we can obtain a data series $D(S_p, q, i)$ ($p = 1, \cdots, K$) of the node N_i at q^{th} period T_q. We suppose that the data sequence $D(p, q, i)$ ($p = 1, \cdots, K$) obey some kind of distribution $\mathcal{N}(t)$ relative to time t. For convenience, we denote the sampled or received data as $D(p, q, i)$ and the aggregated data on node N_i as $D_f(p, q, i)$ according to a kind of aggregation function $\mathbb{f}$, *i.e.*

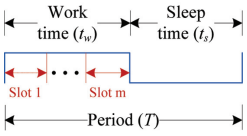
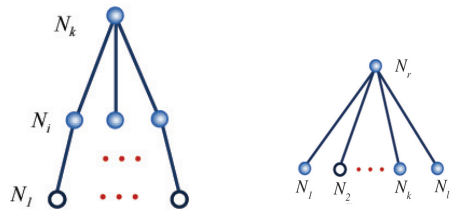


FIGURE 1
Duty cycle



A data aggregation tree DT_k A layer of DT_k

FIGURE 2
A data aggregation tree DT_k. The hollow nodes are leaves.

$D_f(S_p, T_q, i) = \mathbb{f}(D(S_p, T_q, \mathscr{S}))$, where $\mathscr{S} \subset V$ is a subset containing nodes in the aggregation tree parented at N_i. $\mathbb{f}$ can mean the calculation Max, Min, Average and so on. In brief, we use D_f instead of $D_f(S_p, T_q, k)$.

In this paper, we adopt the following *data aggregation algorithm*: $\mathbb{f}$. Each node N_i obtains k_i ($k_i \leq K$) data after sampling data at time slot t_w in T. And N_i aggregates the k_i data into one. If $N_i \notin \mathcal{S}$, N_i sends its aggregated data to its parent. If $N_i \in \mathcal{S}$, N_i waits till it receives certain amount of data from some nodes. Then N_i aggregates its data and that of its leaf into one data, which is transmitted to the sink by multihop fashion. When the leaf N_i transmits its own aggregated data to the sink through other parent nodes N_j, N_j aggregate their data with the N_i's data.

Suppose that there is a routing protocol, which constructs a *fixed routing* to the sink for each node in the network.

We can use the *aggregation tree* DT_k to describe the process of data aggregation and transmission as shown in Figure 2(a). The *aggregation tree* DT_k denotes a tree rooted at the node N_k and the size of DT_k is denoted as $|DT_k|$, which is known by the parent N_k in our function $\mathbb{f}$. At a period T_q, N_k also knows the number of nodes, which gather data from the physical world.

IV SYNCHRONOUS SAMPLING

In this section, all nodes sample data synchronously. We adopt two ways to reduce time slots and the number of nodes needing to transmit data. One way is *controllable data aggregation* while the other is *uncontrollable data aggregation*.

In the following context, we suppose that the data $D(p, q, i)$, directly gathered from the physical world, obeys the *normal distribution* $N(\mu, \sigma^2)$, where u is the expectation and σ is the variance. And $D(p, q, i)$ and $D(p, q, j)$ are independent from each other when $i \neq j$.

A Controllable Data Aggregation

When the links and the nodes are reliable and the interference among the network can be avoided by a precisely designed schedule, we can assume that there is no data loss occurring. Under the case, the sink can receive the aggregated data $D(p,q,i)$ ($p = 1,\cdots,K$ and $i = 1,\cdots,n$) from all nodes at the period T_q without losing any sampled data. We can theoretically analyze the least number m ($m \geq n$) of nodes needed to retransmit their data to the sink when we guarantee $P(|E-E'| < \wp) = 1-\gamma$, where $\wp$ is a small positive value. So we can design an algorithm to select m nodes to transmit their data to the sink through the relay nodes.

There are two cases. The first is that the sink only collects the data from m nodes when these m nodes sample data in all time slots. The second is that the sink also collects the data of m out of n nodes but each of m nodes only sample data in part of time slots.

When all nodes sample and aggregate data at all time slots, the sink can finally obtain an aggregated data D_f^n. We consider D_f^n as a reference. When the sink randomly and uniformly selects m out of n nodes, we denote the value of the aggregated data at the sink as D_f^m.

Theorem 1. *When there are m ($m < n$) nodes, randomly chosen, sampling and aggregating data and each of m nodes gathers data at all time slots, we have* $P\{|D_f^m - D_f^n| < \wp\} = 1-\gamma$, *where* $\wp = \frac{S \cdot t_{\frac{\gamma}{2}(m-1)}}{\sqrt{m}}$, *S is a standard deviation,* $0 \leq \wp$ *and* $0 \leq \gamma \leq 1$.

Proof. Since there are totally n nodes in the network, the average value D_f^n of their sampled data $D(p,q,i)$ is $D_f^n = \frac{1}{n}\sum_{i=1}^{n} \overline{D}(S_p,T_q,i)$ at the period T_q, where $\overline{D}(S_p,T_q,i) = \frac{1}{K}\sum_{p=1}^{K} D(p,T_q,i)$. According to *the law of large numbers*, we can obtain the following equation:

$$P(|D_f^n - \mu| < \epsilon) \geq 1 - \frac{\sigma^2}{n \cdot \epsilon^2}$$

where $\epsilon > 0$ is small positive number. WSN is a kind of large-scale network, we can assume that n is large enough to make $D_f^n = \mu$ with high probability.

When there are only m nodes, which sample, aggregate and send the data to the sink, the average value D_f^m of their sampled data $D(p,q,i)$ is $D_f^m = \frac{1}{m}\sum_{i=1}^{m} \overline{D}(S_p,T_q,i)$ at the period T_q. Notice that the m nodes are randomly chosen. Then the error between D_f^m and the expectation μ of the data in the physical world can be obtained from the following equation:

$$P\{|\frac{D_f^m - \mu}{\sigma}\sqrt{m}| < c\} = 1-\gamma \tag{1}$$

where $c > 0$ and $1 - \gamma$ is a confidence level. The variance σ of the data in the physical world is usually unknown. But the sink can estimate the variance based on the current data according to the following equation.

$$S = \sqrt{\frac{1}{m-1}\sum_{i=1}^{m}(\overline{D}(S_p, T_q, i) - D_f^m)^2} \tag{2}$$

Based on Equation (2), we can obtain following equation according to Equation (1).

$$P\{|\frac{D_f^m - \mu}{S}\sqrt{m}| < c\} = 1 - \gamma$$

Since $\frac{D_f^m - \mu}{S}\sqrt{m}$ obeys $t(m-1)$ *distribution*[1], we have $P\{|t| < c\} = 1-\gamma$, *i.e.*$P\{|t| \geq c\} = 1 - P\{|t| < c\} = \gamma$. Therefore we have the following equation:

$$P\{|D_f^m - \mu| < \wp\} = 1 - \gamma \tag{3}$$

where $\wp = \frac{S \cdot c}{\sqrt{m}}$ and $c = t\frac{\gamma}{2}(m-1)$. That finishes the proof. □

The parameters $\wp$ and γ are previously set at the sink. The above theorem means that the sink can only receive m nodes' data if the difference $\wp$ with the confidence level $1-\gamma$ is acceptable. So $n-m$ nodes can stop gathering data and save energy. Notice that the $n - m$ nodes may still afford the communication task. We denote the parameters $\wp_k$ and γ_k for an arbitrary parent node N_k. When N_k selects its m_k ($m_k \leq m$) leaf nodes to sample and transmit data, $m_k = \frac{m|DT_k|}{n}$ with probability. According to Equation (3), we can generalize the result in Theorem 1 as the following equation.

$$P\{|D_f^{m_k} - \mu| < \wp_k\} = 1 - \gamma_k \tag{4}$$

Where

$\wp_k = \frac{S_{m_k} \cdot c}{\sqrt{m_k}}$, $c = t\frac{\gamma_k}{2}(m_k - 1)$ and

$S_{m_k} = \sqrt{\frac{1}{m_k - 1}\sum_{i=1}^{m_k}(\overline{D}(S_p, T_q, i) - D_f^{m_k})^2}$.

Furthermore, based on the previous schedule, another schedule to save energy and time is that each node samples data at η out of K time slots at

[1] t-distribution has the probability density function: $f_\gamma(t) = \frac{\Gamma(\frac{\gamma+1}{2})}{\sqrt{\gamma\pi}\Gamma(\frac{\gamma}{2})}(1+\frac{t^2}{\gamma})^{-(\gamma+1)/2}$, where γ is the number of degrees of freedom and Γ is the Gamma function.

the period T_q while the sink still randomly and uniformly chooses m nodes to sample data. At this time, we use $D_f^{k\eta}$ to denote the value of the aggregated data at a node N_k when the node samples only during η time slots and there are m nodes to sample data in the whole network. And D_f^k denotes the value of aggregation data at a node N_k when N_k samples at all time slots. The error between $D_f^{k\eta}$ and D_f^k could be bounded to be at most $\widetilde{\wp}_k$ rooted at a node N_k with a given confidence level $1-\gamma_k$, where $\widetilde{\wp}_k \geq 0$ and $0 \leq \gamma_k \leq 1$.

Lemma 1. *When a node N_k randomly chooses m_k ($m_k \leq m$) children in DT_k, to sample and aggregate data and each of m_k nodes samples data at η ($\eta \leq K$) time slots, we have $P\{|D_f^{k\eta} - D_f^k| < \tilde{\wp}_k\} = 1-\gamma_k$, where $\tilde{\wp}_k = \frac{S_k \cdot t_{\frac{\gamma_k}{2}(m_k-1)}}{\sqrt{m_k}}$, S_k is a standard deviation.*

Proof. When each node N_i gathers data at η out of K time slots at period T_q, we can obtain that the average of the η sampled data is $D_f^{i\eta} = \frac{1}{\eta}\sum_{p=1}^{\eta} D(p, T_q, i)$ at period T_q. And the following equation can be obtained according to Theorem 1:

$$P_i\{|D_f^{i\eta} - \overline{D}_f| < \wp_i\} = 1 - \gamma_i$$

where $\overline{D}_f = \frac{1}{K}\sum_{p=1}^{K} D(p, T_q, i)$, $\wp_i = \frac{S_i \cdot c_i}{\sqrt{\eta}}$ and $c_i = t_{\frac{\gamma_i}{2}}(\eta - 1)$ ($0 \leq \wp_i$ and $0 \leq \gamma_i \leq 1$). Here S_i is described in Equation (5).

$$S_i = \sqrt{\frac{1}{\eta - 1}\sum_{p=1}^{\eta}(D(p, T_q, i) - D_f^{i\eta})^2} \tag{5}$$

When there are only m nodes to sample in the whole network, to aggregate data and to send the data to the sink, the expected number m_k of nodes contained in an aggregation tree DT_k is $\frac{m|DT_k|}{n}$. The average value $D_f^{m_k}$ of their sampled data $D(p, q, i)$ is $\widetilde{D}_f^{m_k} = \frac{1}{m_k}\sum_{i=1}^{m_k} \eta D_f^{i\eta}$ at T_q. So the standard deviation in Equation (2) becomes:

$$S_k = \sqrt{\frac{1}{m_k - 1}\sum_{i=1}^{m_k}(D_f^{i\eta} - \widetilde{D}_f^{m_k})^2} \tag{6}$$

According to Equation (3), the error between $\widetilde{D}_f^{m_k}$ and μ is:

$$P\{|\widetilde{D}_f^{m_k} - \mu| < \wp_k\} = 1 - \gamma_k \tag{7}$$

where $\wp_k = \frac{S_k \cdot c_k}{\sqrt{m_k}}$ and $c_k = t\frac{\gamma_k}{2}(m_k - 1)$. □

Notice that the number p of time slots in each period T_q is usually small, so the confidence level $1 - \gamma_k$ can't be very high at certain error $\wp_k$. When we consider the value $D_f^{n\eta}$ of the aggregated data at the sink when m nodes sample only at η time slots, the error between $D_f^{n\eta}$ and D_f^n could be bounded to be at most $\wp_n$ under the confidence level $1 - \gamma_n$, where $\wp_n \geq 0$ and $0 \leq 1 - \gamma_n \leq 1$.

Lemma 2. *In Lemma 1, when the data is aggregated to the sink, we have* $P\{|D_f^{n\eta} - D_f^n| < \wp_n\} = 1 - \gamma_n$.

In Lemma 1, the aggregation tree DT_k grows into a tree rooted at the sink DT_s when the aggregated data is transmitted to the sink finally. The average value D_f^m of their sampled data $D(p, q, i)$ is $\widetilde{D}_f^m = \frac{1}{m}\sum_{i=1}^m \eta D_f^{i\eta}$ at T_q, so the standard deviation is $\tilde{S}_n = \sqrt{\frac{1}{m-1}\sum_{i=1}^m (D_f^{n\eta} - \widetilde{D}_f^m)^2}$. According to Equation (5), the error between $\widetilde{D}_f^m$ and μ is: $P\{|\widetilde{D}_f^m - \mu| < \widetilde{\wp}_n\} = 1 - \gamma$ where $\widetilde{\wp}_n = \frac{\widetilde{S}_n \cdot c_n}{\sqrt{m}}$ and $c_n = t\frac{\gamma_n}{2}(m - 1)$. At the moment, $\widetilde{D}_f^n = \widetilde{D}_f^m$. Lemma 2 means that the error between the aggregation data obtained from all time slots and that obtained from η time slots is also bounded by the error $\wp_n$ under certain confidence level $1 - \gamma_n$ when m nodes sample.

B Noncontrollable Data Aggregation

In the practical environment, the nodes and the links are unreliable and the interference is unavoidable. Therefore, some packets may be lost during transmission because of the unreliability of nodes or some fault occurring. But the number of tolerable data loss and link disconnection can be theoretically figured out.

In the subsection A, it is controllable whether some nodes need sample data without considering the link or the node reliability. This section is unlike the case of the subsection A. When the packet loss does exist and all nodes sample data at all time slots, we denote the aggregated data at the sink by D_f^x. Surely $x \leq n$.

Theorem 2. *Suppose the packet loss probability P_l happens randomly and uniformly in any time slots and among the communication between any pair of nodes. If each node samples data at all time slots, then a parent node can create out the aggregated data with error no bigger than $\wp_k$ under the confidence level $1 - \gamma_k$ when P_l is not bigger than* $\frac{|DT_k| - m_k}{u_k - v_k + \sum_{j=1}^{v_k} |N_j|}$.

Proof. Without loss of generality, a parent N_k of a tree DT_k has u_k leaf nodes, among which v_k leaves have their own leaf nodes. So the data from $(u_k - v_k)P_l$ nodes may be lost with probability. If we denote the v_k leaves as N_j, $j =$

$1, \cdots, v_k$, the data from $P_l \sum_{j=1}^{v_k} |N_j|$ data nodes may be lost with probability. Therefore, there are only $|DT_k| - (u_k - v_k)P_l - P_l \sum_{j=1}^{v_k} |N_j|$ nodes, which can still send their data to N_k. According to Equation (4), $|DT_k| - (u_k - v_k)P_l - P_l \sum_{j=1}^{v_k} |N_j|$ should not be less m_k under the same $\wp_k$ and $1 - \gamma_k$, where m_k, $\wp_k$ and $1 - \gamma_k$ is determined according to Equation (4). So $|DT_k| - (u_k - v_k)P_l - P_l \sum_{j=1}^{v_k} |N_j| \geq m_k$, *i.e.*

$$P_l \leq \frac{|DT_k| - m_k}{u_k - v_k + \sum_{j=1}^{v_k} |N_j|} \tag{8}$$

□

According to the above theorem, a parent can make decision whether its leaves need retransmit their data or not.

When the data loss probability between any pair of nodes is P_l and we denote the aggregated data at the sink as $\widetilde{D}_f^n$, the following lemma can be obtained under a certain error $\wp_l$ and confidence level $1 - \gamma_l$.

Lemma 3. *Suppose the packet loss probability P_l happens randomly and uniformly in any time slots and among the communication between any pair of nodes. When all nodes transmit their data to the sink, we have $P\{|\tilde{D}_f^n - D_f^n| < \wp\} = 1 - \gamma$ when $P_l \leq 1 - \frac{m}{n}$.*

Proof. When the aggregated data is sent to the sink, $|DT_s| = n$ and $m_k = m$. There is no leaf node so $u_s = v_s$, which has no its own leaf. According to Theorem 2, $P_l \leq \frac{n-m}{\sum_{j=1}^{v_s} |N_j|} = \frac{n-m}{n} = 1 - \frac{m}{n}$, where v_s is the number of the leaf nodes of the sink.

So Equation (3) can still be satisfied according to Theorem 1 when $P_l \leq 1 - \frac{m}{n}$. □

When the permanent fault on the links or the nodes occurs, the topology structure should be reconstructed. The issue remains to be researched in the future work. When the temporary nodes or links fault occur, based on the received data, a parent node can make decision whether the fault branches or leaves need retransmit their data.

When the data loss probability of a leaf node is $P_l > 0$, the leaf node is required to retransmit its data. We suppose the retransmission can be successful with probability P_r and define the aggregated data at the node N_k as $\overline{\overline{D}}_f^k$ after retransmission. We can have the following result in Lemma 4.

Lemma 4. *Suppose that the data loss probability of a leaf node is $P_l > 0$ and the successful retransmission probability is P_r. When $P_l \leq \frac{|DT_k| - m_k}{[u_k - v_k + \sum_{j=1}^{v_k} |N_j|]P_r}$, we have $P\{|\overline{\overline{D}}_f^k - D_f^n| < \wp_r\} = 1 - \gamma$, where $\wp_r$ is a given error bound.*

Proof. If a parent node N_k finds that the received aggregated data can not satisfy Equation (8) when the data loss probability is P_l, N_k lets the nodes, failed to transmit, retransmit their data. $[(u_k - v_k)P_l - P_l \sum_{j=1}^{v_k} |N_j|] \times P_r$ nodes may retransmit their data successfully. So there are $|DT_k| - [(u_k - v_k)P_l - P_l \sum_{j=1}^{v_k} |N_j|] \times P_r$ nodes, which finally can transmit their data successfully. According to Equation (4), $|DT_k| - (u_k - v_k)P_l - P_l \sum_{j=1}^{v_k} |N_j|$ should not be less m_k under the same $\wp_k$ and $1 - \gamma_k$, where m_k, $\wp_k$ and $1 - \gamma_k$ are determined according to Equation (4). So $|DT_k| - [(u_k - v_k)P_l - P_l \sum_{j=1}^{v_k} |N_j|]P_r \geq m_k$, *i.e.*

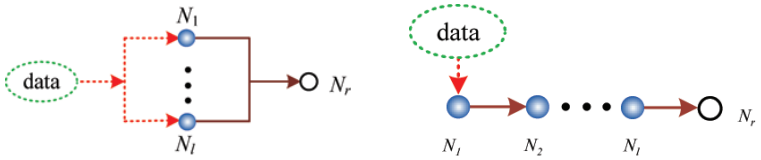
$$P_l \leq \frac{|DT_k| - m_k}{[u_k - v_k + \sum_{j=1}^{v_k} |N_j|]P_r} \tag{9}$$

□

C Confidence Level and Error Allocation

One interesting task is to allocate the error and the confidence level in the network. When the error $\wp$ is acceptable under the confidence level $1 - \gamma$ at the sink, it is necessary to allocate the error and the confidence level at the parent nodes of different sub-trees. For example, if the error $\wp_k$ and the confidence level $1 - \gamma_k$ at the parent N_k are known in Figure 2(a), how does N_k set the error $\wp_i$ and the confidence level $1 - \gamma_i$ for N_i ($N_i \in DT_k$) among DT_k. Here, we discuss the question in the case: without data loss. The case of data loss will be researched in the near future.

Before allocating the error under a certain confidence level among the network, we firstly introduce the *allocation model*. Allocation model is the topology model, based on which, the error under a certain confidence level can be allocated to each parent of its subtree among the network. Generally a tree is constituted by two kinds of basic structures: *SC* and *CC*, as shown in Figure 3. As we know, a WSN collects data from the physical environment and nodes transmits the data to the sink by multihop. Therefore the data always flows unidirectionally from ordinary nodes to the sink. The basic structures



(a) *CC* (b) *SC*

FIGURE 3
Allocation Model. Dotted line means the data is collected by sensors. The solid means that the data is transmitted by radio. The hollow circle is a parent node.

can be modeled as the two kinds of typical circuits. Notice that the links between any pair of nodes are bidirectional.

1) Collateral Connection In Figure 2(a), an aggregation tree rooted at node N_k contains two levels. Each level in a tree contains several leaf nodes. For convenience, we see a layer of an arbitrary aggregation tree DT_k as shown in Figure 2(b). The subtree DT_r rooted at node N_r has l leaves: $N_1, N_2, \cdots, N_l$, where N_2 is also the parents of subsubtree DT_2. Since $N_1, \cdots, N_l$ directly transmit their data to N_r, one node would not affect others' transmission when some of them lose packet. For example, if a packet can not be transmitted from N_1 to N_r, it does not necessarily result in the fault of other transmission from N_2 (or N_3 *etc*) to N_r. Therefore we can consider the subtree DT_r as a parallel system [25] (Figure 3(a)). When we define the confidence level of each node N_i as $1-\gamma_i$ and the error $\wp_r$ under the confidence level $1-\gamma_r$ of the parent is given, the probability P_r that $|D_f^{N_r} - \overline{D}_f| < \wp_r$ is $1-\gamma_r$. So the probability for the parent can be described by the probability of its leaves.

$$P_r = 1 - \prod_{i=1}^{k}(1-P_i) = 1 - \prod_{i=1}^{k}\gamma_i \tag{10}$$

where P_i is the probability about N_i that $P_i(|D_f^{N_i} - \overline{D}_f| < \wp_i)$ is $1-\gamma_i$. Notice that some nodes, such as N_l, is not selected to sample data and N_r only selects k ($k \leq l$) data to sample data in Figure 2(b).

Notice that we are considering the case that there is no data loss happening. Therefore, at least m nodes should be selected to sample and transmit data in the whole network according to Theorem 1. But the selection of the m nodes depends on the error bound and the confidence level given for the sink. It is easy to give the sink the error bound and the confidence level. After that, it is uneasy that the sink allocates its error bound and the confidence level to its leaf nodes and its leaf nodes allocate their error bound and the confidence level to their leaf nodes till all parent nodes are allocated the error bound and the confidence level. In the following context, Theorem 3 gives a method to allocate the confidence level between parents and their leaf nodes.

We design a distributed algorithm as described in Algorithm 1. In the algorithm, we allocate the confidence level $1-\gamma$ pro rata in different nodes when the error $\wp$ is given. In Algorithm 1, the given graph $G(V, \phi)$ composes of the vertex set V and no edge set. $|V| = n$ and the node in V is randomly and uniformly deployed to satisfy the connection condition [23].

Theorem 3. *When no data loss happens and the error bound $\wp_r$ and the confidence level $1-\gamma_r$ for the parent nodes are previously given for the sink, under* CC*, the leaf node N_i of the sink can be allocated the confidence level γ_i, where $\gamma_i = \sqrt[k]{\gamma_i}$ and $k = |DT_s|$.*

Proof. We begin our proof based on the model described in Equation (10). When $k = 1$, $P_r = 1 - \gamma_i = P_i$.

When $k \geq 2$, k, γ_i and $\wp_i$ will be determined in the following context.

There are many important and practical methods to distribute the probability indexes. Here we try to find the minimal error of $\sum_{i=1}^{k} \wp_i$ under certain confidence level P_r while we find a k as small as possible. A smaller k is suitable to save more energy while a bigger k is needed to achieve a lower error $\wp_i$. Here we use *Lagrange undetermined coefficients* method to find k as small as possible while to guarantee the error at a proper level. Here we can construct a *Lagrange function H*:

$$H = \sum_{i=1}^{k} \wp_i + \lambda(P_r + \prod_{i=1}^{k} \gamma_i - 1) \tag{11}$$

In order to find the minimal value of $\wp_i$, we can calculate the derivative of the *Lagrange function H* and let it be zero, *i.e.*, $\frac{\partial H}{\partial \wp_i} = 0$. Since $\wp_i = \frac{S_i \cdot c_i}{\sqrt{k}}$ and $c_i = t_{\frac{\gamma_i}{2}}(k-1)$, we can obtain the below equation:

$$\wp_i = \frac{S_i \cdot t_{\frac{\gamma_i}{2}}(k-1)}{\sqrt{k}} \tag{12}$$

The probability density function of the random variable t is as following:

$$f(t, \nu) = \frac{\Gamma((\nu+1)/2)}{\sqrt{\nu\pi}\Gamma(\nu/2)}(1 + t^2/\nu)^{(\nu+1)/2}$$

where $\nu = k - 1$, $t = \gamma_i/2$ and $\Gamma(z) = \int_0^\infty e^{-t}t^{z-1}dt$.

Based on Equation (11), the partial derivative of H with respect to the variable $\wp_i$ is written as:

$$\frac{\partial H}{\partial \wp_i} = 1 + \lambda \prod_{j=1, j\neq i}^{k} \gamma_j \frac{\partial \gamma_i}{\partial \wp_i} \tag{13}$$

According to Equation (12), we can obtain the partial derivative of γ_i with respect to the variable $\wp_i$.

$$\frac{\partial \wp_i}{\partial \wp_i} = \frac{S_i}{\sqrt{k}} \frac{\partial t \frac{\gamma_i}{2}(k-1)}{\partial \wp_i}$$

$$\Rightarrow 1 = \frac{S_i}{\sqrt{k}} \frac{\partial f(\gamma_i/2, k-1)}{\partial \wp_i} \tag{14}$$

$$\Rightarrow \frac{C}{S_i} = (1 + \frac{\gamma_i^2}{4(k-1)})^{k/2-1} \frac{\gamma_i}{2(k-1)} \frac{\partial \gamma_i}{\partial \wp_i}$$

where $C = \frac{2\sqrt{(k-1)\pi}\Gamma((k-1)/2)}{\sqrt{k}\Gamma((k)/2)}$. Let $\frac{\partial H}{\partial \wp_i} = 0$ in Equation (13) and based on Equation (14), we can obtain that:

$$0 = 1 + \lambda \frac{1 - P_r}{\gamma_i} \frac{\partial \gamma_i}{\partial \wp_i}$$

$$\Rightarrow \lambda = -\frac{\gamma_i}{(1 - P_r)\frac{\partial \gamma_i}{\partial \wp_i}}$$

$$= -(1 + \frac{\gamma_i^2}{4(k-1)})^{k/2-1} \frac{\gamma_i^2 S_i}{2C(k-1)(1-P_r)}$$

Notice that the above equation is tenable for each γ_i ($i = 1, \cdots, k$). So we can obtain the following equation when $i \neq j$.

$$(4(k-1) + \gamma_i^2)^{k/2-1} \gamma_i^2 S_i = (4(k-1) + \gamma_j^2)^{k/2-1} \gamma_j^2 S_j \tag{15}$$

According to Equation (10) and (15), γ_i ($i = 1, \cdots, k$) can be solved out. From Equation (15), we can obtain that:

$$(k/2-1)\ln(4(k-1) + \gamma_i^2) + 2\ln\gamma_i + \ln S_i$$

$$= (k/2-1)\ln(4(k-1) + \gamma_j^2) + 2\ln\gamma_j + \ln S_j \qquad i \neq j$$

Since the k is same for all nodes in the same CC model DT_r, the $\gamma_i = \gamma_j$ when $i \neq j$ based on above equation. According to Equation (10), γ_i can be obtained as following equation.

$$\gamma_i = \sqrt[k]{1 - P_r} = \sqrt[k]{\gamma_r} \tag{16}$$

□

Serial Connection In Figure 2(a), the node N_k is the parent of N_i and N_i is the parent of N_1. There are other leaves at both of parents N_k and N_i. And under

the kind of topology structure, the transmission on one edge affects that on the other. For example, a successful transmission from N_1 to N_k necessarily means that both of transmission from N_1 to N_i and from N_i to N_k should be successful. So the kind topology structure can be modeled as the serial connection in Figure 3(b). When we define the confidence level of each node N_i as $1-\gamma_i$ and the error $\wp_i$ under the confidence $1-\gamma_r$ of the parent is given, the probability P_r that $|D_f^{N_r} - \overline{D}_f| < \wp_r$ is $1-\gamma_r$. The probability for the parent can be described as following equation:

$$P_r = \prod_{i=1}^{k} P_i = \prod_{i=1}^{k} (1-\gamma_i) \tag{17}$$

where P_i is the probability about N_i that $P_i(|D_f^{N_i} - \overline{D}_f| < \wp_i)$ is $1-\gamma_i$.

This paper presents an algorithm (Algorithm 1) to allocate the confidence level from the parent nodes to leaves under both *CC* and *SC* in Figure 3(b). We also gives a theorem (Theorem 4) to describe the feasible of the algorithm.

Algorithm 1 Confidence Level Allocation

Input: A given graph $G(V,\phi)$ and an error bound $\wp$ and a confidence level $1-\gamma$;
Output: A connected tree with each parent node N_i allocated a confidence level $1-\gamma_i$.

1: Use the algorithm in [4] to construct a CDS $\mathcal{S}$;
2: Give the error bound $\wp$ and the confidence level $1-\gamma$ to sink;
3: **if** A node N_i ($\in \mathcal{S}$) has more than one child **then**
4: N_i allocates the confidence level among its children nodes according to Equation (16);
5: **else**
6: N_i broadcasts a message $SerialAllocation_i$, which contains the node ID and a counter $Count_i$. And set $Count_i = 1$;
7: **end if**
8: When a node N_j receives a message $SerialAllocation_i$, N_j sets $Count_i$+=1;
9: **if** N_j has more than one child node *or* is a leaf node **then**
10: N_j sends a message $ReturnSerialAllocation_i$ to N_i, which contains the N_j's ID and all ID in $SerialAllocation_i$ and $Count_i$;
11: **else**
12: N_j keeps on sending $SerialAllocation_i$ to its children;
13: **end if**
14: **if** N_i receives a message $ReturnSerialAllocation_i$ **then**
15: It allocates the confidence level among the nodes contained in $ReturnSerialAllocation_i$ according to Equation (21);

16: It sends a message $AllocationResult_i$ to the nodes contained in $ReturnSerialAllocation_i$.
17: **end if**
18: When N_j receives $AllocationResult_i$, N_j sets its error bound and confidence level according to the results in $AllocationResult_i$;
19: **if** N_j has more than one child **then**
20: Go to step 4;
21: **end if**

Theorem 4. *When no data loss happens and the error bound $\wp_r$ and the confidence level $1-\gamma_r$ nodes are previously given for the sink, under* SC*, the leaf node N_i of the sink can be allocated the confidence level γ_i, where $\gamma_i = \sqrt[k]{P_r}$ and $k = |DT_s|$.*

Proof. We give our proof based on the model described in Equation (17). Here we also try to find the minimal error of $\sum_{i=1}^{k} \wp_i$ under certain confidence level P_r while we find a k as small as possible.

When $k = 1$, $P_r = 1 - \gamma_i = P_i$. When $k \geq 2$, k, γ_i and $\wp_i$ will be determined in the following context.

Firstly we can construct a *Lagrange function H*:

$$H = \sum_{i=1}^{k} \wp_i + \lambda(P_r - \prod_{i=1}^{k} P_i) \tag{18}$$

In order to find the minimal value of $\wp_i$, we can calculate the derivative of the *Lagrange function H* and let it be zero, *i.e.*, $\frac{\partial H}{\partial \wp_i} = 0$.

Based on Equation (11), the partial derivative of H with respect to the variable $\wp_i$ is written as:

$$\frac{\partial H}{\partial \wp_i} = 1 + \lambda \prod_{j=1, j\neq i}^{k} P_j \frac{\partial \gamma_i}{\partial \wp_i} \tag{19}$$

Since $P_i = 1 - \gamma_i$, $\frac{\partial P_i}{\partial \wp_i} = -\frac{\partial \gamma_i}{\partial \wp_i}$. Let $\frac{\partial H}{\partial \wp_i} = 0$ in Equation (19) and based on Equation (14), we can obtain that:

$$0 = 1 + \lambda \frac{P_r}{P_i} \frac{\partial \gamma_i}{\partial \wp_i} \Rightarrow \lambda = -\frac{P_i}{(P_r)\frac{\partial \gamma_i}{\partial \wp_i}}$$

$$= -(1 + \frac{\gamma_i^2}{4(k-1)})^{k/2-1} \frac{\gamma_i P_i S_i}{2CP_r(k-1)}$$

Notice that the above equation is tenable for each γ_i ($i = 1, \cdots, k$). So we can obtain the following equation when $i \neq j$.

$$(4(k-1) + \gamma_i^2)^{k/2-1}\gamma_i P_i S_i = (4(k-1) + \gamma_j^2)^{k/2-1}\gamma_j P_j S_j \tag{20}$$

According to Equation (17) and (20), γ_i ($i = 1, \cdots, k$) can be solved out. From Equation (20), we can obtain that:

$$\begin{aligned}&(k/2-1)\ln(4(k-1)+\gamma_i^2) + \ln\gamma_i P_i + \ln S_i \\ &= (k/2-1)\ln(4(k-1)+\gamma_j^2) + \ln\gamma_j P_j + \ln S_j \qquad i \neq j\end{aligned}$$

Since the k is same for all nodes in the same *SC* case, the $\gamma_i P_i = \gamma_j P_j$ based on above equation when $i \neq j$. According to Equation (17), γ_i and P_i can be obtained as following equation.

$$\gamma_i = 1 - P_i = \sqrt[k]{P_r} \tag{21}$$

□

Now we can design an algorithm to positively control whether a node samples data or not as shown in Algorithm 2. Based the error bound $\wp_r$ at the sink and the confidence level allocation obtained on Algorithm 1, Algorithm 2 makes each parent node know the number of its leaf nodes to sample.

Algorithm 1 Control Data Aggregation

Input: The error bound $\wp_r$ at the sink and the confidence level allocation obtained in Algorithm 1.
Output: The needed number of sampling leaf nodes of each parent node.

1: The sink calculates the needed number of sampling nodes according to the given error bound $\wp_r$ and the confidence level $1 - \gamma$ according to Equation (3);
2: Based on the confidence level allocated in Algorithm 1 and its error bound, each parent node N_i calculates the needed number of sampling nodes according to Equation (4);
3: All parent nodes finish calculating the needed number of sampling data.

D Energy Saving

Here we consider the energy saving under both cases of *controllable* and *uncontrollable data aggregation*.

When we consider that there is no data loss as subsection A, the reason of the energy saving is mainly the data transmission reduction, *i.e.* some nodes

need not sample and transmit the data and others need not sample data under some time slots. Although some packets are inevitably retransmitted because of collision in wireless channel, it has less retransmission to adopt lossy data aggregation schedule since there is less data to transmit.

We define that the energy cost to sample data in a slot as E_s and the energy cost to transmit a packet in one hop as E_p.

Lemma 5. *When no data loss happens and only m nodes gather data at η time slots out of K, the saved energy E_a is $(n-m)((1-\eta)E_s + E_p)$.*

Proof. There are $n-m$ nodes, which can save energy since they need not sample and transmit data. The total saved energy E_t mainly contains two parts: sampling energy and transmitting energy. Notice that each node N_i samples data and transmits its data to other node N_j. N_j aggregates the N_i's data and that of itself into one packet. The energy to transmit a packet in one hop is saved. $E_a = (n-m) \times (1-\eta) \times E_s + (n-m) \times E_p = (n-m)((1-\eta)E_s + E_p)$. □

When each transmission is unreliable with probability P_l and only m nodes gather data, the saved energy E_a is $(n-m)E_p$. Since some data is unavoidably lost, the error under the same confidence level is increased. When we positively argue to retransmit the lost data, the costed energy is $m \times P_l \times E_p$ since the number of lost packets is expectably $m \times P_l$. Under the case, the save energy is $\max\{0,\ (n-m)((1-\eta)E_s + E_p) - m \times P_l \times E_p\}$.

V PERFORMANCE EVALUATION

Simulations for the performance evaluation of our method are conducted with the OMNeT++ simulation tool [26].

In the simulation, this paper considers two cases: with and without packet loss. The packet loss occurs when including the MAC layer in the simulation model. All nodes are deployed in a $1000 \times 1000\ m^2$ area. Variable numbers of nodes from 100 to 1000 are deployed in the area in steps of 100. Here MAC layer implements the 802.11 protocol [27], which adopts CSMA/CA. The receive sensitivity of the radio is at least -98 *dBm*. The antenna can be adjusted for a range of output power levels from -20 *dBm* to 5 *dBm* in steps of 1 *dBm*. The maximal transmission radius is $r_{max} = 260\ m$. The detailed values of the relative parameters are provided in [28] and [29]. In the simulation, we set the sample period $1s$ with duty circle 50%. Each period is divided into 20 time slots. The nodes in the networks sample data from the information in each period. In our simulation, we denote the original information from the environment by the random data obeying Gaussian distribution $N(\mu, \sigma^2)$, where $\mu = 30$ and $\sigma = 5$.

A additional task is to guarantee the connectivity of the original. Esfahanian and Hakimi described some relative algorithms solved the connectivity problem in detail [30]. Esfahanian proposed an algorithm to calculate the

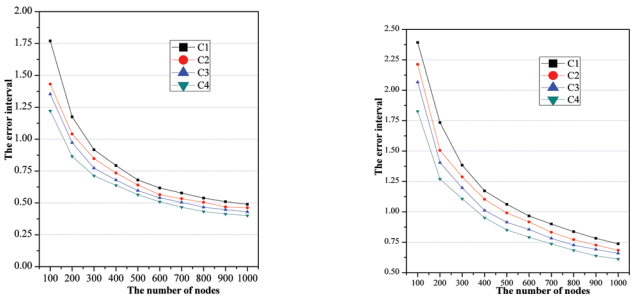
vertex connectivity, which requires $[n - \delta - 1 + \frac{1}{2}k(2\delta - k - 3)]$ times calls of the MFA [31].

A Without Packet Loss

Under this case, all links has no packet loss. We set C1=70%, C2=80%, C3=90% and C4=100%. The error intervals are illustrated respectively in Figure 4(a) and Figure 4(b) while the given confidence levels are respectively equal to 0.8 and 0.9. In both of the figures, the error intervals dramatically decrease when the number of nodes increases. The error interval can be very small when the number of nodes is large enough. In other words, the lossy data aggregation scheme would not incur large error when the number of nodes is large. Therefore it is feasible to adopt lossy data aggregation method in WSNs, especially when the total number of nodes is large.

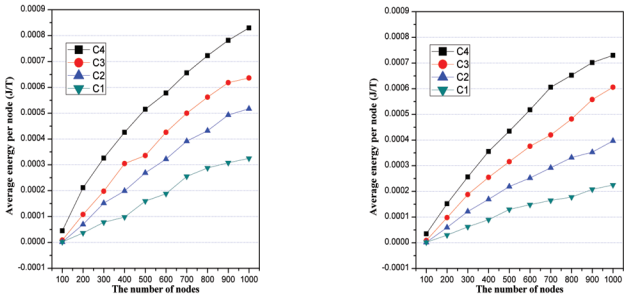
From Figure 4(a), we can also find that the error intervals of C1, C2, C3 and C4 are gradually close to each other when the number of node is increasing. Therefore it would not greatly affect the data aggregation accuracy to adopt lossy data aggregation scheme in WSNs, especially when the total number of nodes is large. The similar result can also be found in Figure 4(b).

In the above simulation, we count the energy consumption under the confidence levels 0.8 and 0.9 in each period. The energy consumptions per node under two cases are respectively presented in Figure 5(a) and Figure 5(a). It can be easily found that there is minimal energy consumed under the case C1 while there is maximal energy consumed under the case C4. With the increasing number of nodes, the difference of energy consumption among four cases C1, C2 and C3 and C4. *i. e.*, it can save more energy when lower percentage of nodes sample data.



(a) $1 - \gamma = 0.8$ (b) $1 - \gamma = 0.9$

FIGURE 4
The error inteval under different given confidence level. C1,C2,C3 and C4 respectively represent:70%, 80%, 90% and 100% nodes to sample data.



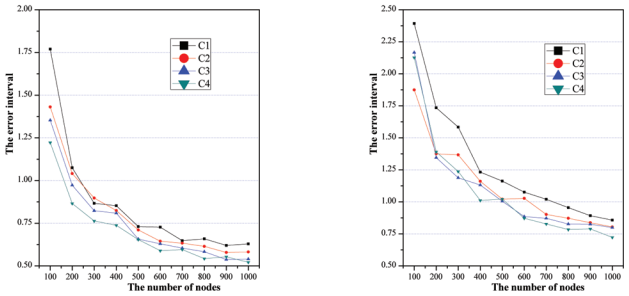
(a) $1-\gamma=0.8$ (b) $1-\gamma=0.9$

FIGURE 5
The energy consumption under different given confidence level.

From both of Figure 4 and Figure 5, we can find that it can save more energy by adopting lower percentage of nodes to sample data and the amount of saved energy can relatively increase when the number of nodes is increasing. At the same time, the data accuracy, which is indicated by the error interval, can be bounded, especially when the number of nodes goes to larger.

B With Packet Loss

Under the case, the link is not reliable, *i.e.*, the packet may be lost because of the wireless interference and media access competition. The error intervals under the confidence levels 0.8 and 0.9 are respectively presented in Figure 6(a) and Figure 6(b). In Figure 6(a), the error interval decreases with the increasing of the number of nodes. Comparing to the error interval in Figure 4(a), the error interval in the figure is higher except the case C1. It is caused by the unpredictable data loss. Under C1, the error interval decreases

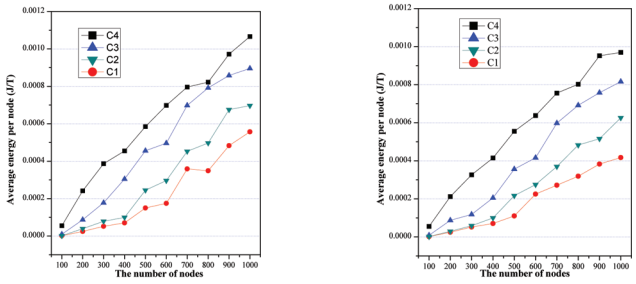


(a) $1-\gamma=0.8$ (b) $1-\gamma=0.9$

FIGURE 6
The error interval under different given confidence level.

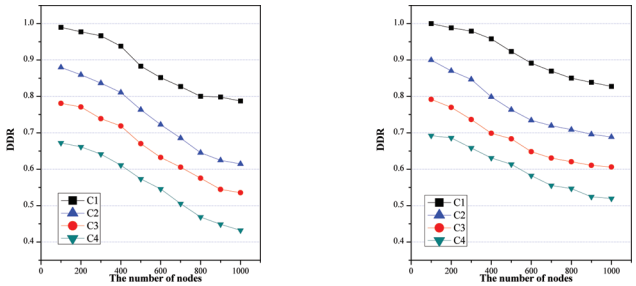
from Figure 4(a) to Figure 6(a), which is also caused by the unpredictable data loss. It results in the smaller difference among the error intervals under C1, C2, C3 and C4. Since the data loss is unavoidable in practical application, it may result in less effect on the data accuracy than that under the case in Figure 4(a). The similar results can also be concluded in Figure 6(b).

Because of the data loss, the energy consumption in Figure 7(a) is higher than that in Figure 5(a). But it is same in both figures that the less number of nodes to sample data causes less energy consumption and the saved energy relatively increases with the increasing of number of nodes. In Figure 8, the effect of confidence level on the data deliver rate (DDR) is presented. Higher confidence level needs higher DDR. And the DDR decreases when the number of nodes increases, which is caused by two reasons. One is the packet loss while the other is that the number of sampled data increases with the increasing of the number of nodes.



(a) $1-\gamma=0.8$ (b) $1-\gamma=0.9$

FIGURE 7
The energy consumption under different given confidence level.



(a) $1-\gamma=0.8$ (b) $1-\gamma=0.9$

FIGURE 8
DDR under different given confidence level.

In both of Figure 5 and Figure 7, we can find that lower confidence level causes higher energy consumption. Lower confidence level means the corresponding γ is high then the error interval is lower, which means that it need more sample data to keep the same probability that the data error falls in a more narrow interval. So a higher confidence level does not mean a good choice. Especially when the number of the nodes is larger, a high confidence level contributes a little to decreasing the error interval. However, a higher confidence level would not necessarily lead to larger difference among error intervals, as show in Figure 6. In practical applications, how to choose the confidence level or the error interval depends on the constraints or the requirements of the users. If a certain error interval is acceptable, it is feasible and energy saving to adopt the scheme in this paper. In fact, the data redundancy often exists in WSNs.

VI CONCLUSION

In this paper, we first study the error between the value obtained from all nodes and that from a part of nodes is bounded when assuming that there is no data loss. We give detailed analysis on the relation between the error bound and the number of sampling nodes or time slots when the confidence level is previously given. In order to minimizing the number of sampling leaf nodes and time slots, we design algorithms to assign the confidence levels among the parent nodes based on corresponding confidence level. We also study the case when data loss exists and compute the probability bound when the confidence level and the error bound are given.

There are some work waiting to be solved in future. In the real network, the DT is often composed of the *SC* and *CC*. It is complex to build the model for the compound structure. When the distribution of the sampled data does not obey Gaussian distribution, we will estimate the probability density function in a complex and time-variation environment. Since the paper has analyzed the error bound under the certain confidence level when the data loss probability $P_l > 0$, we will design the algorithm that allocates the error bound and the confidence level with the existence of the data loss probability. We will analyze the error and confidence level allocation under the asynchronous case. Since the clock offset in WSNs is unavoidable and relatively large [32] and it costs much extra energy to synchronize time in the network.

VII ACKNOWLEDGEMENT

The research of the paper is partially supported by National Natural Science Foundation of China under Grant No.608683126, No.60773042, National Basic Research Program of China (973 Program) under No.2010CB328100,

Science and Technology Planning Program of Zhejiang Province under Grant No.2009C31046.

REFERENCES

[1] Deborah Estrin. Wireless sensor network, part IV:sensor network protocols. Technical report, Mobicom, 2002.

[2] L. Krishnamachari, D. Estrin, and S. Wicker. The impact of data aggregation in wireless sensor networks. In *ICDCS'02*, pages 575–578, 2-5 July 2002.

[3] Z. Eskandari, M.H. Yaghmaee, and A.H. Mohajerzadeh. Energy efficient spanning tree for data aggregation in wireless sensor networks. In *Proceedings of 17th International Conference on Computer Communications and Networks (ICCCN '08)*, pages 1–5, 3-7 Aug. 2008.

[4] P.-J. Wan C.-H. Huang and F. Yao. Nearly constant approximation for data aggregation scheduling in wireless sensor networks. In *IEEE INFOCOM'07*, 2007.

[5] Bo Yu, Jianzhong Li, and Yingshu Li. Distributed data aggregation scheduling in wireless sensor network. In *IEEE INFOCOM'09*, 19th-25th April 2009.

[6] R. Rajagopalan and P.K. Varshney. Data aggregation techniques in sensor networks: A survey. *IEEE Communications Surveys and Tutorials*, 8(4):2–17, 2006.

[7] Y. Wang, W.Z. Wang, and X.Y. Li. Distributed low-cost backbone formation for wireless ad hoc networks. In *ACM Mobicom'05*, pages 2–13, 2005.

[8] Yingshu Li, My T. Thai, Feng Wang, Chih-Wei Yi, Peng-Jun Wan, and Ding-Zhu Du. On greedy construction of connected dominating sets in wireless networks: Research articles. *Wirel. Commun. Mob. Comput.*, 5(8):927–932, 2005.

[9] YanWei Wu, Xiang-Yang Li, YunHao Liu, and Wei Lou. Energy-efficient wake-up scheduling for data collection and aggregation. *IEEE TPDS, accepted to appear*, 2009.

[10] XiaoHua Xu, ShiGuang Wang, XuFei Mao, ShaoJie Tang, and XiangYang Li. An improved approximation algorithm for data aggregation in multi-hop wireless sensor networks. In *FOWANC workshop of ACM Mobihoc, accepted*, 2009.

[11] L. Wang Z.-Y. Wan P.-J. Wan, C.-H. Huang and X. Jia. Minimum-latency aggregation scheduling in multihop wireless networks. In *ACM MOBIHOC 2009*, 2009.

[12] Yi Hu, Nuo Yu, and Xiaohua Jia. Energy efficient real-time data aggregation in wireless sensor networks. In *IWCMC '06*, pages 803–808, 2006.

[13] Yang Yu, Bhaskar Krishnamachari, and Viktor K. Prasanna. Energy-latency tradeoffs for data gathering in wireless sensor networks. In *IEEE INFOCOM'04*, 2004.

[14] Z. Ye, A.A. Abouzeid, and J. Ai. Optimal policies for distributed data aggregation in wireless sensor networks. *IEEE TPDS*, (5):1494–1507, 2009.

[15] T. He, B.M. Blum, J.A. Stankovic, and T. Abdelzaher. AIDA: Adaptive application-independent data aggregation in wireless sensor networks. *ACM Transactions on Embedded Computing Systems (TECS)*, 3(2):426–457, 2004.

[16] YunHao Liu YanWei Wu, Xiang-Yang Li and Wei Lou. Energy-efficient wake-up scheduling for data collection and aggregation. *IEEE TPDS*, 10 Mar. 2009.

[17] Gregory Hartl and Baochun Li. Loss inference in wireless sensor networks based on data aggregation. In *IPSN '04*, pages 396–404. ACM, 2004.

[18] Huang Lee and Abtin Keshavarzian. Towards energy-optimal and reliable data collection via collision-free scheduling in wireless sensor networks. In *IEEE INFOCOM'08*, pages 2029–2037, 2008.

[19] D. Kim, Y. Wu, Y. Li, F. Zou, and D.Z. Du. Constructing minimum connected dominating sets with bounded diameters in wireless networks. *IEEE TPDS*, pages 147–157, 2009.

[20] D. Li, H. Du, P.J. Wan, X. Gao, Z. Zhang, and W. Wu. Construction of strongly connected dominating sets in asymmetric multihop wireless networks. *Theoretical Computer Science*, 410(8-10):661–669, 2009.

[21] Jie Wu, Mihaela Cardei, Fei Dai, and Shuhui Yang. Extended dominating set and its applications in ad hoc networks using cooperative communication. *IEEE TPDS*, 17(8):851–864, 2006.

[22] Toshihiro Fujito and Hiroshi Nagamochi. A 2-approximation algorithm for the minimum weight edge dominating set problem. *Discrete Appl. Math.*, 118(3):199–207, 2002.

[23] M. D. Penrose. The longest edge of the random minimal spanning tree. *The Annals of Applied Probability*, 7(2):340–361, may 1997.

[24] Peng-Jun Wan, Khaled M. Alzoubi, and Ophir Frieder. Distributed construction of connected dominating set in wireless ad hoc networks. *Mob. Netw. Appl.*, 9(2):141–149, 2004.

[25] Charles Ebeling. *An Introduction to Rliablity and Maintainability Engineering*. The McGraw-Hill Companies, 1997.

[26] OMNeTpp. http://www.omnetpp.org/.

[27] Information technology-telecommunication and information exchange between systems-local and metropolitan area networks-specific requirements-part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications. *IEEE Standard, Technical Report.*, 1999.

[28] MICA2 Datasheet. http://www.xbow.com/.

[29] MPR/MIB Node Hardware Users Manual. http://www.xbow.com/.

[30] A. H. Esfahanian. *On the evolution of connectivity algorithms,invited, submitted, Selected Topics in Graph Theory*. Cambridge University Press, 2007.

[31] A. H. Esfahanian and S. L. Hakimi. On computing the connectivities of graphs and digraphs. *Networks*, pages 355–366, 1984.

[32] Benjamin R. Hamilton, Xiaoli Ma, Qi Zhao, and Jun Xu. ACES: adaptive clock estimation and synchronization using kalman filtering. In *ACM MOBICOM'08*, pages 152–162, 2008.