Energy-Efficient Data Acquisition in Wireless Sensor Networks through Spatial Correlation

Li Xun and Tang Shiqi  
School of Mechatronics Engineering and Automation  
National University of Defense Technology  
Changsha, Hunan Province, China  
{lixun, tangshiqi}@nudt.edu.cn  

Geoff V Merrett and Neil M White  
School of Electronics and Computer Science  
University of Southampton  
Southampton, UK  
{gvm,nmw}@ecs.soton.ac.uk

Abstract - The application of Wireless Sensor Networks (WSNs) is restrained by their often-limited lifetime. A sensor node's lifetime is fundamentally linked to the volume of data that it senses, processes and reports. Spatial correlation between sensor nodes is an inherent phenomenon to WSNs, induced by redundant nodes which report duplicated information. In this paper, we report on the design of a distributed sampling scheme referred to as the 'Virtual Sampling Scheme (VSS)'. VSS is formed from two components: an algorithm for forming virtual clusters and a distributed sampling method. VSS primarily utilizes redundancy of sensor nodes to get only a subset to sense the environment at any one time. Sensor nodes that are not sensing the environment are in a low-power sleep state, thus conserving energy. Furthermore, VSS balances the energy consumption amongst nodes by using a round robin method.

Index Terms - Virtual cluster; wireless sensor networks; spatial correlation.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) consist of multiple sensor nodes [1] deployed over an area for monitoring some phenomena or tracking targets. In target tracking WSNs, sensor nodes cooperate to acquire the signal of the spatially and temporally dynamic target. Generally, a sensor node is powered by a local battery which has a limited energy budget. There is a contradiction for a sensor node that it should have a sufficient lifetime to finish the application task, while recharging the battery can be difficult, costly and even impossible in a hostile environment. Although a sensor node can be powered by energy harvesting technology [2], the energy obtained from the external environment is dependent on the environmental conditions and is rarely continuous.

Energy conservation is an important research field for WSNs. Anastasi et al. classified the majority of energy conservation techniques into duty cycled and data-driven approaches [3]. Communication protocols, such as S-MAC [4], commonly adopt a duty cycled approach. When communication is not required, a sensor node switches into a low-power sleep state to reduce the energy consumption. Recent research has shown that some types of sensors can consume more energy than the communication process [5]. Data-driven approaches to conserve energy can eliminate unnecessary sensing by adjusting the sample frequency and reducing the number of sensor nodes which participate in sensing. Methods of energy conservation often aim to put more sensor nodes into a sleep state, but tracking applications usually require a certain number of sensor nodes to be activate in order to sufficiently sample the target signal.

Generally, sensor nodes are densely deployed. Therefore sensor data from adjacent nodes will be highly redundant. Consecutive samples from the same node are also likely to exhibit temporal correlation. Vuran et al. analyzed the spatial and temporal correlation of WSNs [6]. Reducing redundant data can conserve a node's energy. Redundant information is not useful to data fusion, but sensing, processing and communication of the information can consume extra unnecessary energy.

Eliminating the acquisition of redundant data will have a significant effect on nodes in a WSN. The energy consumed on data acquisition and on communication will be both reduced. Without redundant data, the data processing is expedited in the microcontroller and a lower data rate will be created. So the occupation of the wireless channel and the probability of inducing congestion will be lower. As a result, the lifetime of WSNs is prolonged, and the QoS is increased.

In this paper, we investigate the energy conservation and balance (i.e. the distribution of energy throughout the nodes in the network) of WSNs. We present a novel scheme, referred to as Virtual Sampling Scheme (VSS), to fulfill distributed data acquirement in WSNs. VSS is based on virtual clusters (VCs), which represent a set of adjacent nodes. We design an algorithm to form VCs, which derives the VCs by using a spatial correlation model before nodes start to sample the target signal. The algorithm operates without calculating a complex distortion function as required by other reported research. The computation required in our algorithm is also lighter. We present a round robin sampling scheme as the second part of VSS, which balances the energy consumption of each sensor node in a VC. Therefore VSS assures the accuracy in signal estimation. The VSS algorithm is simulated to evaluate the fusion accuracy and energy consumption.

II. RELATED WORK

In a WSN, multiple sensor nodes often sample a signal at the same time. The node to fuse the sensor data from nodes may get more consecutive sensor data in a fusion periods. Two different methods can be used for data fusion, as Figure 1 shows:
A. Networks architecture

WSNs may have a flat or hierarchical architecture. A hierarchical WSN consists of clusters of nodes, where each cluster has a cluster head (CH) responsible for communicating with other clusters. In this paper, a hierarchical WSN with \( N \) sensor nodes is considered. Nodes \( n_1, n_2, \ldots, n_N \) are stationary in a set \( G \). Node \( n_i \) (\( i = 1, 2, \ldots, N \)) is at location \( s_i(x_i, y_i) \). All sensor nodes are divided into \( M \) clusters, every CH is resource-rich nodes which can perform data aggregation as presented in [10]. Let \( C_j \) (\( j = 1, 2, \ldots, M \)) denote the \( j \)th cluster. The CH of \( C_j \) is \( ch_j \) and the number of nodes (including \( ch_j \)) in \( C_j \) is \( M_j \), defined by equation (1).

\[
\sum_{j=1}^{M} M_j = N
\]

\[
C_j = \{ n_i, ch_j | n_i \in G, i = 1, 2, \ldots, M_j - 1 \} \tag{1}
\]

B. Spatial correlation model

Spatial correlation between two nodes is decided by the distance between them. Berger et al. discussed four spatial correlation models [10]. The power exponential model is often used in applications, and defined in equation (2).

\[
K(d_{ij}) = e^{-d_{ij}/\theta_1}, \theta_1 > 0 \tag{2}
\]

Where \( d_{ij} \) is the distance between node \( n_i \) and node \( n_j \), calculated as equation (3).

\[
d_{ij} = \| s_i - s_j \| = (x_i - x_j)^2 + (y_i - y_j)^2 \tag{3}
\]

At each observation point, the signals, from the source signal which is modeled as Gaussian random variables, are joint Gaussian random variable [6]. The noise is modeled as independent and identically distributed Gaussian random variables. Sensor data, combining signal with noise, can be obtained using equation (4).

\[
X_i = S_i + N_i \tag{4}
\]
Where $S_i = (0, \sigma_{S_i}^2)$ and $N_i = (0, \sigma_{N_i}^2)$. The covariance of sensor data of nodes $n_i$ and $n_j$ is given by equation (5).

\[
Cov(X_i, X_j) = Cov(S_i + N_i, S_j + N_j) = E(S_i S_j) + E(S_i N_j) + E(N_i S_j) + E(N_i N_j)
\]

\[
= E(S_i S_j) = \sigma_i^2 e^{-d_{ij}/\theta}.
\]

If $d_{ij}$ is small enough, $Cov(X_i, X_j)$ will be relatively large. $X_i$ and $X_j$ are highly redundant, nodes $n_i$ and $n_j$ do not need to join the sensing process together.

IV. VIRTUAL SAMPLING SCHEME

A. Definition of virtual cluster

Definition: A node set (called $VC_{ij}$) which has $M_{ij}$ nodes is called a ‘Virtual Cluster’ of cluster $C_i$, if the following conditions are satisfied.

1) If $n_k \in VC_{ij}$, then: $n_k \in C_i, k = 1, 2, \cdots M_{ij}$

2) If $n_k, n_l \in VC_{ij}, k, l = 1, 2, \cdots M_{ij}$, then $|s_k - s_l| \leq d_{\text{min}}$

Where $d_{\text{min}}$ is a distance threshold. A VC should have a head (VCH) which implements a scheduling function. Assume that the VC number in $C_i$ is $K_i$, the VCH of $VC_{ij}$ is $vch_{ij}$.

\[
VC_{ij} = \{n_i, vch_{ij}, d_{\text{min}} | n_j \in C_i, l = 1 \cdots M_{ij} - 1\}
\]

\[
\sum_{j=1}^{K_i} M_{ij} = M_i, \quad i = 1, 2 \cdots M, \quad j = 1, 2 \cdots K_i
\]  

(6)

Figure 2 presents the relationship between cluster and VCs.

B. Forming a virtual cluster

A VC is a set of nodes with a VCH and a distance threshold. In order to form a VC, we should set a VCH, derive $d_{\text{min}}$ and find nodes with the limitation $|s_k - s_l| \leq d_{\text{min}}$. A VC can be any shape in topology, and can be thought of as a circle whose center is VCH and radius is $d_{\text{min}}/2$. All nodes belonging to the VC are in the circle and satisfy $|s_k - s_l| \leq d_{\text{min}}$. It is enough that only one sensor node in this VC performs sampling function at a time, other nodes can enter into sleep state.

Figure 3 shows an example of VCs, where nodes 1, 8, 15, 23, 26 and 30 are VCHs. The values $r$ and $d$ denote the communication radius of CH and circle radius of VCs, respectively.

![Virtual clusters](image)

The process of forming VCs starts from the CH. After the cluster is constructed, the process can begin. CH sets distance threshold and sends it to other nodes.

Assumption 1: Every node in $G$ has an exclusive identification (ID) number. Some clustering algorithms are based on the node’s ID number [11], so the assumption is easy to satisfy.

Assumption 2: Every node can calculate the distance to other nodes. Guoqiang Mao et al. presented some method about how to calculate the distance between nodes [12].

Assumption 3: The expected distance threshold is shorter than the communication radius of a node, as shown in Figure 3. The assumption can ensure the correlation amongst nodes is relatively high.

If a node in the cluster $C_i$, assume that it is $n_k$, wants to form a VC, it should broadcast a message with the expected distance value $d$ and its ID. Node $n_k$ will become a VCH after it broadcasts the message. Node $n_i$, which is still not in a VC, will be at one of three possible states after $n_k$ broadcasts its message. The three possible states are as follows.

State 1: $S1 \rightarrow |s_k - s_l| \leq d$

State 2: $S2 \rightarrow |s_k - s_l| \leq r, |s_k - s_l| > d$
State3: \( S3 \rightarrow \|s_k - s_i\| > r \)

If node \( n_i \) receives the broadcast message, it calculates the distance \( \|s_k - s_i\| \) and compares it with \( d \). Node \( n_i \) will join the VC if it is in state S1 after comparison. It will reply to \( n_k \) with its ID. If node \( n_i \) is in state S2, it will set a timer \( T_1 \) with random time once it receives the broadcast message. Before \( T_1 \) expires, \( n_i \) should keep listening to a new message. If it can change to state S1, it will send a reply message and join the VC that the VCH is the node which sends the broadcast message. When \( T_1 \) expires, \( n_i \) sends a broadcast message to claim a new VC. Node \( n_i \) will be a VCH.

Initially, all nodes are in state S3, every node sets a timer \( T_2 \) with random time, which should be longer than \( T_1 \). Before \( T_2 \) expires, a node should listen to a broadcast message and change state to S1 or S2. When timer \( T_2 \) of a node expires, the node can not receive a broadcast message, and it does not know value \( d \). E.g., In Figure 2, although node 30 is close to node 29, but node 29 can not broadcast a claiming message after it joins a VC, so node 30 can not receive any broadcast message. Once \( T_2 \) expires, node 30 broadcasts an enquiring message with its ID. Node 29 will receive the enquiring message and send a reply with value \( d \). After receiving the value \( d \), node 30 broadcasts a message to claim a new VC. Figure 4 presents the states relationship. In Figure 4, state S5 means that the node has been a member of one VC. S4 is a middle state. In S4, a node will send a broadcast message to become a VCH after it receives a reply for its enquiry.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{states.png}
\caption{States relationship in one node}
\end{figure}

When a VC has been formed, the VCH should maintain a ID table. Every node in a VC will receive some parameters from their VCH, and this VCH should report its ID to the CH. Each VCH will receive some parameters from a CH and sends these parameters to each node, such as the sample frequency.

There is no central node to control the forming process, it is distributed. The CH will be the first node to broadcast message. Note that a node in state S5 will not reply to any broadcast message from other nodes to claim a new VC. Figure 5 describes the algorithm running at a node.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{algorithm.png}
\caption{Forming algorithm}
\end{figure}

C. Distributed round robin sampling in a virtual cluster

In a VC, all nodes are equal. Every node can perform sampling function as the representative of the VC. We can set a node to sample the signal while other nodes can work in sleep state. But sampling node will consume its energy more quickly than others. We adopt a distributed round robin sampling scheme to balance energy amongst sensor nodes.

While forming a VC, the VCH receives the reply from other nodes with their ID, and knows the total number of nodes and their ID. Assume that the ID table is \( \{id_1, id_2, \ldots, id_M\} \) of \( VC_{ij} \), ordered from small to large. Every ID has an index in the table, \( \text{Index}(id_k) = k \). The round robin scheme requires all nodes in the VC to synchronize with \( vch_{ij} \). Because there are only limited nodes in a VC, the reference broadcast synchronization method [13] can be used. The distributed sampling process has two steps: allocation and start, both steps are scheduled by \( vch_{ij} \).
Allocation: \( vch_y \) allocates sample time for all nodes. The round robin period is \( T = M_y / f_{\text{min}} \). The \( id_k \) node samples the source signal at time:

\[
t_0 + \frac{k}{f_{\text{min}}} + T \times m, \quad (m = 1, 2, \cdots)
\]

(7)

Where \( t_0 \) is the start time to sample, \( f_{\text{min}} \) is the expected sample rate. In this step, \( vch_y \) sends parameters \( T \), \( k \) and \( f_{\text{min}} \) to every node in the same VC.

Start: This step starts the sampling process. \( vch_y \) sends a start command with starting time \( t_0 \), every node start to sample according to equation (7).

Figure 6 shows the sampling and sleeping period of nodes.

![Sampling scheme diagram](image1)

Figure 6. Sampling scheme

We use the same method discussed in [6] to fuse the data and analyze the tracking error. Minimum mean square error estimation is used to estimate the event \( S_i \) signal. The estimation, \( Z_i \), is:

\[
Z_i = \frac{E(S_i X_i)}{E(X_i^2)} X_i = \frac{\sigma_S^2}{\sigma_S^2 + \sigma_N^2} X_i = \frac{\sigma_S^2}{\sigma_S^2 + \sigma_N^2} (S_i + N_i)
\]

(8)

The estimation of \( S \), also the fusion result, is:

\[
\hat{S} = \frac{1}{K} \sum_{i=1}^{K} Z_i = \frac{\sigma_S^2}{\sigma_S^2 + \sigma_N^2} \sum_{i=1}^{K} X_i = \frac{\sigma_S^2}{\sigma_S^2 + \sigma_N^2} \sum_{i=1}^{K} (S_i + N_i)
\]

(9)

The distortion between the source and the fusion result is:

\[
D = E[(\hat{S} - S)^2] = \sigma_S^2 + \frac{\sigma_S^2}{K(\sigma_S^2 + \sigma_N^2)} \sum_{i=1}^{K} \rho_{i,j} + \frac{\sigma_N^2}{K(\sigma_S^2 + \sigma_N^2)} \sum_{i=1}^{K} \sum_{j=1}^{K} \rho_{i,j} - \frac{2\sigma_S^2}{K(\sigma_S^2 + \sigma_N^2)} \sum_{i=1}^{K} \rho_{i,j}^2
\]

(10)

Where \( \rho_{i,j} \) is the spatial correlation coefficient between source and nodes \( n_i \), \( \rho_{i,j} \) is the spatial correlation coefficient between nodes \( n_i \) and \( n_j \).

D. Analysis of energy conservation

In every \( T \) interval, a node is active for \( 1 / f_{\text{min}} \) time to sample the signal. Assume that the consuming power of sensing unit of a sensor node in active mode and sleep state is \( P_A \) and \( P_S \), respectively. The energy consumption of a node is:

\[
E_N = P_A \times \frac{1}{f_{\text{min}}} + P_A \times (T - \frac{1}{f_{\text{min}}}) = \frac{1}{f_{\text{min}}} \times \left[ P_A + P_A \times (M_y - 1) \right]
\]

(11)

If the round robin sampling scheme is not adopted, a node should be active in the whole \( T \) interval, the conservation of energy of a node is:

\[
1 - \frac{E_N}{P_A \times T} = (1 - \frac{1}{M_y}) \times (1 - \frac{P_S}{P_A}) = 1 - \frac{1}{M_y}
\]

(12)

Only a node is active in a VC within \( 1 / f_{\text{min}} \) interval, the VC will consume energy:

\[
E_Y = P_A \times \frac{1}{f_{\text{min}}} + P_A \times (M_y - 1) \times \frac{1}{f_{\text{min}}}
\]

(13)

If the round robin scheme is used, the ratio of which a VC conserves energy is:

\[
1 - \frac{E_Y}{P_A \times M_y \times \frac{1}{f_{\text{min}}}} = (1 - \frac{1}{M_y}) \times (1 - \frac{P_S}{P_A}) \approx \frac{1}{M_y}
\]

(14)

From equations (12) and (14), we can conclude that a node conserves energy as the same level as the VC which the node belongs to. The ratio of energy conservation is only related with the number of nodes in the VC. The number of nodes is larger, the energy conservation is more.

V. SIMULATION RESULTS

The simulation environment has 100 nodes and covers a \( 50 \times 50 \) m\(^2\) area. Each node is randomly deployed in this area. The radius of a VC is set as 10m. The result of simulation shows fifteen VCs are formed if the algorithm illustrated in Figure 5 is used. Figure 7 illustrates the fifteen VCHs and, as an illustration, the location and distribution of VCs one to five.

![VCH and VCs distribution](image2)

Figure 7. Distribution of VCH and VCs

There is only one source signal in the simulation. The signal is a Gaussian random variable with expectation and variance of \( (0, 1) \). The noises at each observed location are joint Gaussian random variables with expectations and variances of \( (0, 0.01) \). By using the fusion method given in equation (9), Figure 8 shows the fusion result and the variance of the fusion result at a fusion node. The number of samples is 100.
referred to as VSS. VSS is a novel method to acquire data in a
This paper presents a distributed virtual sampling scheme,
set the radius of the VC as 5m and 10m, respectively.
In Figure 8a, a few points in Figure 8b show a greater variance, but in most cases, although only a subset of the nodes take part in the sampling process at any one time, the fusion result is improved.
In Figure 9, the energy conservation of the VCs is shown. The ratio of energy conservation of a VC is calculated as equation (14) shows. In order to compare different VCs, we set the radius of the VC as 5m and 10m, respectively.
The ratio of energy conservation is proportional to the number of nodes in a VC. As the radius of a VC increases, the number of nodes inside the VC also increases, and hence permits better energy conservation (as each node spends more of its time in a sleep state). Figure 9a shows the largest ratio is 95%, while the largest ratio is 90% in Figure 9b. Some VCs, such as VC14 in Figure 9a, has only one node, the node will continue to sample the signal and can not conserve energy according to VSS. Because the spatial correlation model is related to the radius, careful consideration should be taken in the selection of the radius. A larger radius can give better energy conservation, though may result in worse fusion. As proved in section III, each node in the VC consumes the same amount of energy. Therefore, Figure 9 is also showing the energy conservation of the individual nodes.

VI. CONCLUSION

Based on spatial correlation among adjacent sensor nodes, this paper presents a distributed virtual sampling scheme, referred to as VSS. VSS is a novel method to acquire data in a WSN. In order to realize VSS, we design a round robin sampling scheme and an algorithm to form the VCs. Through analysis and simulation, it is proved that VSS can conserve node’s energy while also providing a better fusion result. By VSS, each node in a VC consumes the same amount of energy, hence balancing energy consumption.

VSS can reduce redundant sensor data to conserve energy, while retaining the meaningful information. However, if a VC has few nodes, the nodes in this VC will consume energy at a faster rate. This raises a significant challenge, and we are currently investigating refinements to VSS to balance the energy between different VCs. Furthermore, while VSS should ensure the accuracy of the fusion result, our simulation results have shown that, in a small number of cases, VSS can give a worse result. By combining the spatial correlation with the number of nodes, we aim to find the optimal node set to get a better fusion result while keeping a high efficiency of energy conservation.

REFERENCES