Data Compression Algorithm based on Hierarchical Cluster Model for Sensor Networks

Wang Lei, Wang Tongsen, Yang Ronghua
Department of Electronic Information & Electrical Engineering
Fujian University of Technology,
Fuzhou, P.R.China, 350108

Abstract

A new distributed algorithm of data ccompression based on hierarchical cluster model for sensor networks is proposed, the basic ideas of which are as follows, firstly the whole sensor network is mapped into a kind of hierarchical clusters model, and then different wavelet transform models are used to commit data compression in inner and super clusters respectively, according to the relative regularity of sensor nodes deployed in the inner clusters, and the relative irregularity of sensor nodes deployed in super cluster. Theoretical analyses and simulation results show that, the above new methods have good performance of approximation, and can compress data and reduce the amount of data efficiently. So, it can prolong the lifetime of the whole sensor network to a greater degree.

1. Introduction

With capabilities of sensing, data processing, GPS positioning and communicating, sensors have great potential in a wide variety of commercial and military applications including environmental and military monitoring, earthquake and weather forecast, underground, deep water and outer space exploration[1-5]. Due to the surrounding uncertainty, hundreds and thousands of sensor nodes need to be deployed simultaneously, and those nodes are self-organized to collect special information of the sensory field. So the studies of sensor networks consisted with large number of sensors (sensor nodes) are attracting intense interests for both academia and industries, and are regarded as an challenging research topic in the 21st century [6].

The main purpose of sensor networks is to collect original data of monitored objects, but a large number of sensor nodes may generate a mass of original data in the whole network, which will exceed the transmission ability of the sensor network because the sensor nodes possess limited energy and transmission bandwidth. So the collected original data will need to be compressed firstly in the in-network before being sent to the sink nodes.

Research results show that data compression can prolong the lifetime of sensor networks efficiently. But compared with traditional Ad-Hoc network, the WSN has characteristics such as sufficiently dense distribution, lack of backbone network, independent topology, limited energy, transmission bandwidth, computation and storage capability and etc. Therefore, new methods shall be introduced for sensor networks to commit the data compression.

It is an important research topic for scientists to study how to reduce the amount of data generated in the in-network efficiently, prolong the lifetime of the whole sensor network remarkably, and decrease the transmission delay at the same time. Since wavelet transformation is such a kind of mathematical tool, which can represent the real and imaginary parts of the signals, has multi-resolution analysis characteristics, and can maintain the statistical peculiarity of signals in different scales or compression ratios. Therefore, it sounds to be a practical way to deal with the large amount of data generated in the in-network by using wavelet transformation.

Currently, some basic research works have been done as to the studies of data compression based on wavelet transformation for sensor networks. For instance, Ciancio and Ortega [7] proposed a distributed data compression algorithm based on lifting wavelet, in which correlation of data can be reduced through information exchanges between neighboring nodes. Since it is designed for the single-hop self-organizing networks and requires that all sensor nodes can send data directly to the sinks, which is contradicted with the limited transmission ability of sensor nodes, therefore, this kind of algorithm cannot be used actually in many real applications. In 2004, Chen and RACE [8] proposed another kind of data compression algorithm based on haar wavelet with rate adaptivity and error bound for sensor networks, which can be used for compression of time-series signals generated by single sensor node, and can reduce the redundant data to be transmitted efficiently through mining the time correlation of data. In 2005, Acimovic, Cristescu, and Lozano [9] proposed two kinds of new distributed data collection algorithms based on haar wavelet too, in which, the first algorithm adopts haar wavelet to commit data compression during the period when data is transmitted to sink nodes, and the second one adopts self-adaptive algorithms to divide sensor nodes into groups at first, and then adopt haar wavelet to commit data compression in each group. In 2006, Xie and Wang [10] proposed a kind of new data compression model based on interval wavelet transformation, in which some nodes are selected to form a backbone of the whole network firstly, and then data are compressed by using interval wavelet transformation when being sent to the sinks along the path found in the backbone.

One of the shortages of all those above algorithms is that these kinds of algorithms adopt simple and regular wavelet transformations such as haar wavelet, lifting wavelet and so on only, which may guarantee the simplicity and feasibility of these algorithms because of the simple infrastructures of haar wavelet and lifting wavelet etc, but at he same time, will make these algorithms difficult to meet the requirements of different applications also. The second shortage is that neither of those above data compression algorithms considered both the time correlations and the space correlations between neighboring nodes simultaneously. So the real applications of those above algorithms are limited to some degree by these two shortages.

In order to overcome these two shortages, a kind of novel data compression algorithm based on hierarchical cluster model for sensor networks is proposed in this paper, in which the whole sensor network is mapped into a kind of hierarchical cluster model consisted with inner and super clusters firstly, and then interval wavelet transformation that is suitable for regular occasions is adopted to commit data compression in the inner clusters because sensor nodes deployed in the inner clusters are distributed with relative regularity, and multi-resolution wavelet transformation that is suitable for irregular occasions is adopted to commit data compression in the supper clusters according to the relative irregularity of sensor nodes deployed in super cluster. Theoretical analyses and simulation results show that our new methods have good performance of approximation, and can compress data and reduce the amount of data to be transmitted efficiently. So, it can prolong the lifetime of the whole sensor network to a greater degree, and is a kind of efficient and practical data compression algorithm for sensor networks.

The paper is organized as follows. Section 2 introduces concepts of first order radio model and hierarchical cluster model of sensor networks. In section 3, we proposed the data compression algorithm based on the hierarchical cluster model. In section 4, we conduct a wide range of simulation experiments to testify the good compression performance of our new methods, and finally we conclude the paper in section 5.

2. first order radio Model and hierarchical cluster Model of sensor networks

2.1. First Order Radio Model

In this paper, the first order radio model [11] of sensor networks will be used to analyze the energy consumption of sensor networks, in which the energy consumed by transmitting *k* bits data through distance *d* can be represented by the following equation:

Transmit energy consumption $E_{Tx}(k,d)$:

$$E_{Tx}(k,d)= \delta_{elec} *k+ \xi_{amp} *k*d^2 \qquad (1)$$

Transmit energy consumption $E_{Rx}(k)$:

$$E_{Rx}(k)= \delta_{elec} *k \qquad (2)$$

Where δ_{elec} is the coefficient of energy consumption of receiver and transmitter when a sensor node are receiving or transmitting data, ξ_{amp} is the coefficient of energy consumption of channel transmission when data are transmitted through wireless channel from the source node to a destination node.

According to the above first order radio model, sensor nodes will consume energy in the processes of both receiving and transmitting data , so the energy consumption of both receiving and transmitting data shall be minimized when we design the data compression algorithms.

2.2. Hierarchical Cluster Model

Definition 1(Cluster): Supposing that all sensor nodes in the sensor network have the same communication radius *r*, and for each sensor node *i*, assuming that its coordinate is (x_i,y_i), then *i* belongs to cluster (m_i,n_i) if and only if the following formulas are satisfied: $m_i =[x_i/ \frac{r}{\sqrt{2}}]$, $n_i =[y_i/ \frac{\mathrm{r}}{\sqrt{2}}]$, where "/" is the division operator, and "[*x*]" means Ceiling(*x*).

For example: Supposing that *r*=15, and the coordinate of sensor node *i* is (40, 40), then we can obtain that m_i=4, n_i=4, therefore, node *i* will belongs to cluster (4, 4). Additionally, assuming that the coordinate of sensor node *j* is (35, 37), then we can obtain that m_j=4, n_j=4, therefore, node *j* will belongs to the same cluster (4, 4) also.

From definition 1, it is easy to know that the following theorems can be drawn.

Theorem 1: Supposing that all sensor nodes in the sensor network have the same communication radius *r*, there exists a unique way of cluster division that satisfies the formulas given in definition 1 for any sensor nodes.

Theorem 2: Any two sensor nodes situated in a same cluster are neighbouring nodes.

Proof: It's east to know that a cluster is a square area with length $\frac{r}{\sqrt{2}}$ and width $\frac{r}{\sqrt{2}}$ from definition 1, so the maximum distance between any two nodes in a cluster is $\sqrt{(\frac{r}{\sqrt{2}})^2+(\frac{r}{\sqrt{2}})^2} = r$, which means the two nodes are situated in the communication radius of each other. Therefore, the conclusion of theorem 2 is proved to be correct.

Supposing that there are N sensor nodes in the sensor network totally, and before deployment, all the sensor nodes are encoded according to the coding scheme of the n-dimensional hypercube nodes[12-13] such as $j_1j_2...j_n$, where $j_1, j_2,..., j_n \in \{0,1\}$, n=[ln$N$]. And in addition, for each sensor node i, assuming that its coordinate is (x_i,y_i), and all sensor nodes in the sensor network have the same communication radius r, and the whole sensor network form a connected graph after all the sensor nodes are deployed, then we can present our automated generation algorithm of hierarchical cluster model as following by combing those above assumptions and the rules of cluster division described in definition 1:

Algorithm 2.1(Automated Generation Algorithm of Hierarchical Cluster Model, AGAHCM):
1.Each sensor node i calculates it cluster number that it belongs to according to the definition 1, where the clusters generated called inner clusters.
2.For each inner cluster A, supposing that all of the nodes that belong to A in the whole sensor network are $\{i_1, i_2,..., i_n\}$, then $i_1, i_2,..., i_n$ can form a ring L_A according to the order of their serial numbers, since all the nodes are encoded according to the coding scheme of the n-dimensional hypercube nodes previously.
3.All inner clusters will form super clusters ultimately according to the following automated generation algorithm of super clusters called AGASC.

Before the algorithm AGASC is presented, the requirements for clustering will be analyzed according to the characteristics of sensor networks at first.

Obviously, since the sensor nodes in the sensor network have those characteristics such as sufficiently dense distribution, lack of backbone network, independent topology, limited energy, transmission bandwidth, computation and storage capability and etc. Therefore, the clustering algorithms designed for sensor networks specially shall satisfy these following requirements:
1) The algorithms shall be distributed, since the sensor nodes in the sensor network are densely distributed.
2) The selection of cluster heads shall adopted the mechanism of the rotation of nodes in order to prolong the lifetime of the whole network. And additionally, the following 2 kinds of factors shall be considered at the same time:
2.1) the number of cluster heads in the areas that sensors are densely deployed shall be larger than that in the areas that sensors are sparsely deployed.
2.2) the nodes with lower average energy consumption and higher residual energy shall have more chances to become cluster heads than those with higher average energy consumption and lower residual energy.

According to the above analyses, we can set the threshold $V(i)$ for each sensor node i as following formula (3):

$$V(i)=\begin{cases} D(i): if \quad r=1, \\ \left[\dfrac{P}{1-P(R \bmod \dfrac{1}{P})} \right] * D(i) * E(i): if \quad i \in G \quad and \quad r \geq 2, \\ 0: otherwise \end{cases} \qquad (3)$$

Where, P represents the percentage of cluster heads in the total nodes of the whole network, R represents the rounds of selecting cluster heads anew, G represents the node set consisted with nodes that are not be selected as cluster heads in the latest 1/P rounds, $D(i)$ is a kind of density-adjust function that can be calculated by the following formula (4), and $E(i)$ is a kind of energy-consumption-adjust function that can be calculated by the following formula (5).

$$D(i)=Nodedensity(i) \,/\, (Nodedensity(i)+1) \qquad (4)$$

Where $Nodedensity(i)$ represents the number of neighboring nodes of node i.

$$E(i)=F(\frac{E_{avg}(i)}{E_{cur}(i)})+\left(R_s\, div \frac{1}{P}\right)\left((1-F(\frac{E_{avg}(i)}{E_{cur}(i)})\right) \qquad (5)$$

Where, $F(\frac{E_{avg}(i)}{E_{cur}(i)})=\begin{cases}1-\dfrac{E_{avg}(i)}{E_{cur}(i)}:if\ E_{avg}(i)\le E_{cur}(i)\\ 0:otherwise\end{cases}$,

and R_s represents the rounds that node i is not selected as a cluster head continuously, and once the node i is selected as a cluster head, then R_s will be set to 0. $E_{cur}(i)$ represents the residual energy of node i, and $E_{avg}(i)$ represents the average energy consumption of all past rounds.

Definition 2(Distance): For any inner cluster A, supposing that all of the nodes that belong to A in the whole sensor network are $\{i_1, i_2,..., i_n\}$, $i_1, i_2,..., i_n$ form a ring L_A according to the order of their serial numbers, then the distance between node j and ring L_A is defined as: $dist(j, L_A)=\min\{d(j, i_1), d(j, i_2),\dots, d(j, i_n)\}$, where $d(j, i_k)$ is the Euclidean distance between node j and i_k.

According to the formula (3) and definition 2, we can present the automated generation algorithm of super clusters called AGASC as following:

Algorithm 2.2(Automated Generation Algorithm of Super Clusters, AGASC):

1.For any sensor node i, if $i\in G$, then i calculates formula (3) independently to obtain the threshold $V(i)$.

2.Node i generates a random number Num(i) that is between 0 and 1.

3.If $V(i)\geq$Num(i), then node i will be selected as a super cluster head, and broadcast the information such as its election, location and serial number etc.

4.For any inner cluster A, supposing that all of the nodes that belong to A in the whole sensor network are $\{i_1, i_2,..., i_n\}$, $i_1, i_2,..., i_n$ form a ring L_A according to the order of their serial numbers, then $i_2,..., i_n$ select the super cluster head that is nearest to ring L_A as their own super cluster head, and then join in the super cluster that the super cluster head generates. (If there are more than one super cluster heads that have the same nearest distance to ring L_A , then $i_2,..., i_n$ select the one with the highest residual energy and smallest serial number as their own super cluster head).

5.For any inner cluster A, supposing that all of the nodes that belong to A in the whole sensor network are $\{i_1, i_2,..., i_n\}$, and there does not contain a super cluster head in $\{i_1, i_2,..., i_n\}$, then the node with the highest residual energy and smallest serial number will be selected as an inner cluster head of inner cluster A from $\{i_1, i_2,..., i_n\}$.

6.For any node p in the inner cluster A, if p is not an inner cluster head, or a super cluster head, and the set of its neighbouring nodes consisted with nodes that are not in the same inner cluster of p contains nodes that do not belong to the set of neighbouring nodes of a cluster head y (y may be an inner cluster head or a super cluster head) of cluster A, which is consisted with nodes that are not in the same inner cluster of y, then p will be selected as a gateway.

7.Steady working period.

8.Every time cycle T, turn to step 1 to re-select super cluster heads to form super clusters.

Definition 3(Core): The node set $C\subseteq V$ in graph G is called a core if and only if $\forall\, p\in V \Rightarrow p\in C$ or p is a neighboring node of some node q that belongs to set C.

Definition 4(Connected Core): Given a graph G=(V, E), if the node set $C \subseteq V$ in graph G satisfies that the sub-graph educed by C is connected and C is a core of G, then C is called a connected core of G.

Theorem 3: If the whole sensor network is regarded as a connected undirected graph G=(V, E), where V is the node set consisted with all nodes in the whole sensor network, E is the channel set consisted with all wireless channels in the whole sensor network, then the node set Ψ ={p| $p\in V$ and p is a super cluster head, an inner cluster head, or a gateway} obtained by algorithm AGAHCM is a connected core of graph G.

Proof: Firstly, from theorem 2 and the step 5 of algorithm AGASC, we can know that for each node $i\in V$, i is an inner cluster head, or a super cluster head, or i is neighbouring to an inner cluster head or a super cluster head , which means there exists at least a super cluster head or an inner cluster head in i's neighbouring nodes. So, it is obvious that Ψ is a core of graph G. Secondly, we will prove that Ψ is connected through the method of induction.

Since all sensor nodes have the same communication radius r, then we can assume that r=1 for convenience sake. Therefore, $\forall$ p, $q \in \Psi$ and p, q are super cluster heads or inner cluster heads, the distance $d(p,q)$ between p and q can be represented as the hops of the shortest path between p and q. It's obvious that $d(p,q)<R$(R is a limited integer) since graph G is connected.

(1)If $d(p,q)=1 \Rightarrow p,q$ are neighboring nodes $\Rightarrow$ p,q can communicate with each other directly$\Rightarrow$ Ψ is connected.

(2) If $d(p,q)=2\Rightarrow$ there exists a path (p,r,q) in graph G because graph G is connected, and p,q are not neighboring nodes$\Rightarrow$ p,q belong to different inner clusters, and r is neighboring to both p and q $\Rightarrow$ if r is not a super cluster head or an inner cluster head, then there exists a r's neighbouring node q, which is not in the same inner cluster of r, and does not belong to the set of neighbouring nodes of p too. Therefore, we can know that r must be a gateway from the step 6 of algorithm AGASC, which means that $r\in \Psi \Rightarrow \Psi$ is connected.

(3) If $d(p,q)=3\Rightarrow$ there exists a path (p, r_1, r_2,q) in graph G because graph G is connected, and p,q are not neighboring nodes$\Rightarrow$ $d(p,r_2)$=2, and $d(r_1,q)$=2.

(3.1) Since $d(p,r_2)$=2 and there exists a path (p, r_1, r_2) in graph G $\Rightarrow$ It's easy to know that r_1 must be a gateway and $r_1\in \Psi$, by simply imitating the previous proof process of (2).

(3.2) Since $d(r_1,q)$=2 and there exists a path (q, r_2, r_1) in graph G $\Rightarrow$ It's easy to know that r_2 must be a gateway and $r_2\in \Psi$, by simply imitating the previous proof process of (2).

Since we have proved above that there are $r_1\in \Psi$ and $r_2\in \Psi$, then we can know that Ψ is connected.

(4) Supposing that Ψ is connected when $d(p,q)=m(m>3)$.

(5) If $d(p,q)=m+1\Rightarrow$ there exists a path (p, r_1, r_2,…, r_m, q) in graph G $\Rightarrow$ $r_2 \in \Psi$ or r_2 is neighboring to some node $r \in \Psi$, since Ψ is a core of graph G $\Rightarrow$ $d(p,r)\leq 3$ and $d(r,q)\leq m$.

(5.1) Since $d(p,r)\leq 3\Rightarrow$ It's easy to know that Ψ is connected, by simply imitating the previous proof processes of (1),(2),and (3).

(5.2) Since $d(r,q)\leq m\Rightarrow$ It's easy to know that Ψ is connected from the previous proof process of (4).

So, from the above proof processes, we can know that Ψ is connected.

3. data compression algorithm based on the hierarchical cluster model

3.1. Data Compression in Inner Clusters

For any inner cluster A, supposing that all of the nodes that belong to A in the whole sensor network are {i_1, i_2,..., i_n}, i_1, i_2,..., i_n form a ring L_A according to the order of their serial numbers, we assume that the order of serial numbers of i_1, i_2,..., i_n satisfy $i_1<i_2<...<i_n$ for

convenience sake. Supposing that the data collected by the kth sensor node i_k in the ring L_A is a kind of time-series signals, which can be regarded as a m-dimensional column vector $C_i = \left(c_{0,i}, c_{1,i}, ..., c_{M-1,i}\right)$, where c_{ki} represents the ith data in the time-series signals collected by the kth sensor node i_k. So the original data collected by the nodes in the ring L_A can be represented as the following matrix C^0:

$$C^0 = (C_1, C_2, \ldots, C_n) = \begin{pmatrix} c_{0,0} & c_{0,1} & \cdots & c_{0,N-1} \\ c_{1,0} & c_{1,1} & \cdots & c_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ c_{M-1,0} & c_{M-1,1} & \cdots & c_{M-1,N-1} \end{pmatrix} \tag{6}$$

Since $i_1, i_2, ..., i_n$ form a ring $L_{A,}$ it is obvious that the first column C_1 is neighbouring to the last column C_n in matrix C^0, which means all the columns C_1, C_2, …, C_n in matrix C^0 form such kind of a ring structure as being illustrated in the following Fig.1.

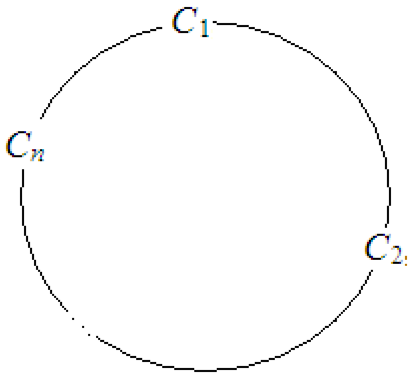

Figure 1. The ring structure of columns C_1, C_2, …, C_n in matrix C^0

From the above Fig.1, we can know that the ring structure of columns C_1, C_2, …, C_n in matrix C^0 seems as a nature period extension of the vector of signals (C_1, C_2, …, C_n), then the boundary effect problem generated by wavelet transformation can be solved naturally[15], if we choose a node in the ring as a beginner, and from which we begin to commit the wavelet transformations.

Till now, we have known that the data stored in the nodes in the ring L_A can be represented as a matrix C^0, and then, the mining of time-correlations of data stored in each node can be mapped into the column wavelet transformation of matrix C^0, and the mining of space-correlations of data stored in the neighbouring nodes in the ring L_A can be mapped into the row wavelet transformation of matrix C^0. It is obvious that the column wavelet transformation can be done in each single node, so it does not need additional communication costs, but the row wavelet transformation needs the information exchanges between neighbouring nodes, so it needs additional communication costs. Therefore, communication costs will be saved if the column wavelet transformations are committed at first.

Let L_n and H_n represent the low-pass filters and high-pass filters of wavelet respectively, then we can obtain the following fast Mallat decomposition algorithm:

$$c_{m,i}^{1,L} = \sum_n L_{n-2m} c_{n,i} \text{ , } 0 \le m < \frac{N}{2} \tag{7}$$

$$c^{1,H}_{m,i} = \sum_{n} H_{n-2m} c_{n,i} \text{ , } 0 \le m < \frac{N}{2} \tag{8}$$

Where $c^{1,L}_{m,i}$ represent the low-frequency wavelet coefficients of the *i*th row and *m*th column, obtained through the first order column wavelet transformation. $c^{1,H}_{i,col}$ represent the corresponding high-frequency wavelet coefficients. Since this kind of wavelet transformations are committed in each single node, so the wavelet coefficients, obtained through the wavelet transformation, can be rearranged. Therefore, the matrix C^0 can be changed into the following matrix C^{01}:

$$C^{01} = \begin{pmatrix} c^{1,L}_{0,0} & c^{1,L}_{0,1} & \cdots & c^{1,L}_{0,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ c^{1,L}_{M/2-1,0} & c^{1,L}_{M/2-1,1} & \cdots & c^{1,L}_{M/2-1,N-1} \\ c^{1,H}_{0,0} & c^{1,H}_{0,1} & \cdots & c^{1,H}_{0,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ c^{1,H}_{M/2-1,0} & c^{1,H}_{M/2-1,1} & \cdots & c^{1,H}_{M/2-1,N-1} \end{pmatrix}$$

The second step is to commit wavelet transformations for the matrix C^{01} to mine the space relativity of data. As for low-pass filters and high-pass filters $L_n$and H_n, the matrix C^{01} will be changed into another matrix C^{02}, after the row wavelet transformations are committed for the matrix C^{01}. Therefore, the following matrix C^1 can be generated by selecting those low-frequency wavelet coefficients from matrix C^{02}:

$$C^{1} = \begin{pmatrix} c^{1,LL}_{0,l_0} & c^{1,LL}_{0,l_1} & \cdots & c^{1,LL}_{0,l_{N/2-1}} \\ c^{1,LL}_{1,l_0} & c^{1,LL}_{1,l_1} & \cdots & c^{1,LL}_{1,l_{N/2-1}} \\ \vdots & \vdots & \ddots & \vdots \\ c^{1,LL}_{\frac{M}{2}-1,l_0} & c^{1,LL}_{\frac{M}{2}-1,l_1} & \cdots & c^{1,LL}_{\frac{M}{2}-1,l_{N/2-1}} \end{pmatrix}$$

Similar to the process of wavelet transformations committed to matrix C^0, wavelet transformations can be committed to the matrix C^1 also. And the rest may be deduced by analogy until the *K*th order spatio-temporal wavelet transformations are committed. Till then, the data are effectively compressed since the spatio-temporal relativities of data are reduced after these above wavelet transformations. After that, these obtained wavelet coefficients can be encoded and sent to corresponding inner cluster heads for farther compression.

In addition, from the algorithm AGASC, we can know that there exists at least one inner cluster head or super cluster head in the ring L_A. So, we can present the data compression algorithm for inner clusters as following.

Algorithm 3.1 (Data Compression Algorithm for Inner Clusters, DCAIC):

1. For any node i_k in the ring L_A, i_k commits wavelet decomposition according to formula (7) and formula (8), in order to reduce the time redundancy of data collected by i_k through mining the time-correlations of data stored in i_k. Supposing that the column vector of wavelet coefficients obtained by the wavelet decomposition is

$$c_k^1 = (c_{1k}^{1L}, c_{2k}^{1L}, ..., c_{[m/2]k}^{1L}, c_{1k}^{1H}, c_{2k}^{1H}, ..., c_{[m/2]k}^{1H})^T ,$$

Where c_{jk}^{1L} and c_{jk}^{1H} represent the low-frequency and high-frequency wavelet coefficients at the jth scale obtained by the above wavelet transformation respectively.

2. If there is a super cluster head in the ring L_A, then the super cluster head will be selected as the ring head of L_A, otherwise the inner cluster head will be selected as the ring head of L_A. And then, all nodes in the ring L_A will send their low-frequency wavelet coefficients obtained by wavelet decomposition to the ring head, so that the ring head can obtain a matrix C_A consisted with the low-frequency wavelet coefficients of all nodes in the ring L_A:

$$C_A=(C_{A1}^L, C_{A2}^L, \ldots, C_{An}^L)=\begin{bmatrix} c_{11}^{1L} & c_{12}^{1L} & \cdots & c_{1n}^{1L} \\ c_{21}^{1L} & c_{22}^{1L} & \cdots & c_{2n}^{1L} \\ \cdots & \cdots & \cdots & \cdots \\ c_{[m/2]1}^{1L} & c_{[m/2]2}^{1L} & \cdots & c_{[m/2]n}^{1L} \end{bmatrix}$$

3. As for the matrix C_A, corresponding row wavelet transformations are committed to reduce the space redundancy between neighbouring nodes in the ring L_A, supposing that the matrix obtained by row wavelet transformations is C_A^1 .

4. As for the matrix C_A^1 , second order column and row wavelet transformations can be committed, and do like this until the Kth-order column and row wavelet transformations are committed, then the original data stored in the nodes of the ring L_A are transformed into the data belong to the wavelet domain. Since the time correlations and the space correlations are wiped off, then much less bits are needed to represent the wavelet coefficients than to represent the original data.

5. After making suitable choices according to the system's requirements, the ring head will send the suitable wavelet coefficients to its super cluster head together with the area of inner cluster A. (The detail method to calculate the area of cluster A will be given in the following sections).

From the algorithm AGAHCM, it is easy to know that there is a little number of sensor nodes included in an inner cluster, since the inner cluster is a very small area. Therefore, we can assume naturally that the sensor nodes are deployed regularly in every inner cluster, which means that we can use these simple interval or haar wavelet transformation in the above algorithm DCAIC to commit data compression for inner clusters.

Additionally, from the step 1 of algorithm DCAIC described above, we can know that the signal extension is not necessary when the interval or haar wavelet transformation is adopted to commit data compression in each single node to reduce the time redundancy of collected data, since the data collected by sensor nodes is regarded as a kind of one dimensional signals.

3.2. Data Compression in Super Clusters

In most of irregular situations, the model of approximation piecewise-constant signal is usually adopted to commit data compression. The approximation piecewise-constant signals are such kind of signals that appear in constant segmentations, and the signal quantity is approximately the same in each segment too. But we can know that the sensor nodes are not deployed regularly in a super cluster from the algorithm AGASC because the density of nodes in each inner cluster are not the same, which leads to the difference of areas covered by all of the nodes in each inner cluster. So the data collected by the nodes in each inner cluster shall be regarded to represent the value of the region covered by the nodes in the inner cluster, which means that we shall establish a kind of corresponding relationship between the data collected by the nodes in each inner cluster and the area covered by the nodes in the inner cluster.

Therefore, we shall consider both the data collected by the nodes in each inner cluster and the area covered by the nodes in the inner cluster, which means that we cannot calculate the *LP* and *H* simply according to the data collected by the nodes in each inner cluster, and we shall consider the area covered by the nodes in the inner cluster at the same time.

Thus, in this section we will propose a kind of new method to commit the data compression for the super clusters based on multi-resolution wavelet transformation, which is suitable for the irregular occasions. Before the actual data compression algorithm for supper clusters is designed, we will firstly give a new calculation method of the area of an inner cluster as following.

First of all, all the nodes in the inner cluster are regarded as discrete growth points. Secondly, Delaunay triangle meshes can be generated by these discrete growth points. Thirdly, Voronoi meshes can be generated according to the Delaunay triangle meshes. Finally, we can generate Voronoi meshes indirectly.

Since these Voronoi meshes generated from the nodes in each inner cluster that is formed through algorithm AGAHCM can be regarded as a kind of local Voronoi meshes of the whole sensor network, then all of these local Voronoi meshes can be merged into a global integrated Voronoi mesh, in which each sensor node can easily calculate the area of its Voronoi cell according to one of the Delaunay edges only. In 1991, Aurenhammer [14] proposed a simple method to calculate Delaunay edge with computation cost of $O(nlgn)$ only for sensor networks with n sensor nodes.

Supposing that all of the nodes that belong to A in the whole sensor network are $\{i_1, i_2,\ldots, i_n\}$, and the areas of Voronoi cells corresponding to $\{i_1, i_2,\ldots, i_n\}$ are $\{S_1, S_2,\ldots, S_n\}$ respectively, then, according to these above descriptions, the area S_A of the inner cluster A can be estimated as:

$$S_A = \sum_{i=1}^{n} S_i \qquad (9)$$

Based on the formula (9), we can present the data compression method for the super clusters as following: At first, the ring head of each inner cluster will send its low-frequency wavelet coefficients and the area of the inner cluster to the super cluster head that it belongs to for next-round of compression, and then the super cluster head will obtain a set consisted with low-frequency wavelet coefficients and areas of its inner clusters. For a super cluster SC, supposing that the low-frequency wavelet coefficient and area of its tth inner cluster t are L_t and S_t respectively, then the super cluster head of SC can calculate its low-frequency and high-frequency wavelet coefficients according the following formula (10) and (11) respectively:

$$LP = \frac{\sum_{t \in SC} S_t L_t}{\sqrt{\sum_{t \in SC} S_t}} \qquad (10)$$

$$HP_t = \frac{\sqrt{S_t}}{\sum_{t \in SC} S_t} \left\{ \sum_{\substack{j=t+1 \\ j \in SC}} S_j L_j + \sum_{\substack{j=t+1 \\ j \in SC}} S_j L_t \right\} \qquad (11)$$

On the basis of the above descriptions, we can give the data compression algorithm for super clusters as following:

Algorithm 3.2 (Data Compression Algorithm for Super Clusters, DCASC):

1. the super cluster head calculates the low-frequency wavelet coefficient *LP* and high-frequency wavelet coefficient HP_t according the following formula (10) and (11) respectively.
2. Return *LP* and HP_t obtained through step1.

3.3. Integrated Data Compression Algorithm based on the Hierarchical Cluster Model

Based on the analyses and descriptions in above sections, we can present the integrated data compression algorithm based on the hierarchical cluster model as following.

Algorithm 3.3 (Data Compression Algorithm based on Hierarchical Cluster, DCAHC):

1. Generating the inner clusters according to the algorithm AGAHCM.
2. Generating the supper clusters according to the algorithm AGASC.
3. Generating the Voronoi meshes according to the methods proposed by Aurenhammer [14].
4. Committing data compression in each inner cluster according to the algorithm DCAIC.
5. Committing data compression in each supper cluster according to the algorithm DCASC.
6. The super cluster heads send their compressed data to sink nodes.

4. Experimental Result

In this section, some experiments are done to verify the data compression performance of our newly proposed spatio-temporal data compression methods for sensor networks. In our experiments, the experimental data are selected from the *Precipitation Dataset* [16], in which, the daily rainfalls of south-north districts within *50km* scope of Pacific Ocean from 1949 to 1994 are provided, and the simulation platform of *SENSE(sensor network simulator and emulator)*[17] is adopted. The detailed experimental steps are as follows:

1) In a 100*100 region, 1000 sensors are generated and form a connected graph. And each sensor node is coded as $\underbrace{(0000000000)}_{10} \sim \underbrace{(1111100111)}_{10}$, just coded as same as the nodes in a hypercube.

2) Then, every node records the daily rainfalls given by *Precipitation Dataset*, and uses the newly proposed algorithm in this paer to do data compression.

3) And finally, an index named Compression Ratio, which is defined as data amount/ bits after data compression, is used to judge the data compression performance of the newly proposed algorithm.

The simulation result is shown in the following Fig.2.

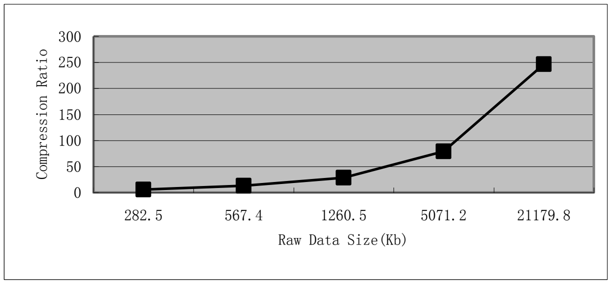

Figure.2. Data compression performance

From Fig.2, it is easy to see that the newly proposed spatio-temporal data compression method in this paper has the good data compression performance.

5. Conclusion And Future Works

A new distributed algorithm of data ccompression based on hierarchical cluster model for sensor networks is proposed, theoretical and experimental results show that the newly proposed method are efficient in prolonging the life cycle of sensor networks, since it has good data compression performance, and can reduce the data transmission in sensor networks to some degree. Multi-wavelets have lots of new characteristics other than the traditional uni-wavelets, so, in the future we will research on how to utilize those excellent characteristics of multi-wavelets to improve our data compression models, in order to improve its data compression performances.

6. Acknowledgement

This work was supported in part by the Natural Science Foundation of Fujian Province of China (No. 2008J0012) and the Key Project of Fujian Science and Technology Foundation of China (No. 2008H0001)

7. Preferences

[1] Gupta I, Riordan D, Sampalli S. Cluster- Head election using fuzzy logic for wireless sensor networks. In: Proc. of the 3rd Annual Communication Networks and Services Research Conf. Halifax: IEEE Computer Society, 2005,3(6):255-260

[2] Megerian S, Koushanfar F, Potkonjak M, Srivastava MB. Worst and best-case coverage in sensor networks. IEEE Trans. on MobileComputing, 2005, 4(1): 84-92

[3] L Wang, ZP Chen, Researches on Scheme of Pairwise Key Establishment for Distributed Sensor Networks ACM Workshop onWireless Multimedia Networking and Performance Modeling(WMuNeP '05) Montreal, Quebec, Canada 2005

[4] L Wang, YP Lin, ZP Chen, A Distributed Data-Centric Clustering Hierarchical Routing Algorithm for Sensor Networks, Acta Electronica Sinica,2004,32(11): 1883 - 1889.

[5] Optimal rate allocation for energy-efficent multipath roting in wireless ad hoc network IEEE Trans. On wireless communications, 2004,3(3):891-899

[6] Estrin,D., Govindan,R., Heideman,J., Kumar,S. Next century challenges: Scalable Coordination in Sensor Networks. In:Proc. of the 5th annual ACM/IEEE international conference on Mobile computing and networking ,Seattle,Washington,1999,pp.263-270

[7] A. Ciancio and A. Ortega, A distributed wavelet compression algorithm for wireless sensor networks using lifting, in Proceedings of the 2004 InternationalConference on Acoustics, Speech and Signal Processing - ICASSP04, Montreal,Canada, May 2004

[8] Chen HM, Li J, Mohapatra P. RACE: Time series compression with rate adaptivity and error bound for sensor networks. In: Proc. of the 2004 IEEE Int'l Conf. on Mobile Ad-Hoc and Sensor Systems. Piscataway: IEEE, 2004. 124.133.

[9] Acimovic J, Cristescu R, Lozano B. Efficient distributed multiresolution processing for data gathering in sensor networks. In: Proc. of the Int'l Conf. on Acoustics, Speech, and Signal Processing. Piscataway: IEEE, 2005. 837.840.

[10] Zhijun Xie, Lei Wang,et.al. An Algorithm of Data Aggregation Based on Data Compression for Sensor Networks. Journal of Software, 2006, 17(4): 860-867

[11] Heinzelman W, Chandrakasan A, Balakrishnan H. Energy-Efficient communication protocol for wireless microsensor networks. In: Hyun-Kook K, ed. Proc. of the Hawaii Int'l Conf. on System Sciences. LNCS 2662, Berlin: Springer-Verlag, 2003. 181.191.

[12] J. Al-Sadi l, K. Day, M. Ould-Khaoua. Probability-based Fault-tolerant Routing in Hypercubes. The Computer Journal, 2001, 44(5): 368-373.

[13] Toshihiko Sasama. On Fault-Tolerance of Hypercubes using Subcubes. International Journal of Reliability, Quality and Safety Engineering, 2002, 9(2): 151-161.

[14] Aurenhammer F. Voronoi diagrams-a survey of a fundamental data structure [J]. ACM Computing sureys.1991.23:345-405

[15] SW Zhou, YP Lin,et.al, A Wavelet Data Compression Algorithm Using Ring Topology for Wireless Sensor Networks, Jpurnal of Software, 2007,18(3): 669-680

[16] M.Widmann and C.Bretherton. 50km resolution daily precipitation for the pacific Northwest,1949-94

[17] Chen, G., J. Branch, M. J. Pflug, L. Zhu, et al. SENSE: A Sensor Network Simulator, Advances in Pervasive Computing and Networking. 2004: 249-267