A Data Aggregation Algorithm Based on Splay Tree for Wireless Sensor Networks

ZHANG Shu-Kui * [1,2], CUI Zhi-Ming[1,2] ,GONG Sheng-Rong[1], LIU Quan[1] ,FAN Jian-Xi[1]
[1]School of Computer Science and Technology, Soochow University, Suzhou, China ,215006
[2] JiangSu Province Support Software Engineering R&D Center for Modern Information Technology Application in Enterprise, Suzhou, China, 215104
Email: {zhangsk, szzmcui, shrgong, liuquan, jxfan }@suda.edu.cn

Abstract-**Detecting the region of emergent events is an important application of wireless sensor networks (WSN). One of the key challenges in detecting event in a WSN is how to detect it accurately while transmitting minimum information to provide sufficient details about the event. In this paper, an aggregation algorithm based on splay tree is proposed to achieve the following goals: monitoring data of any portion of the region can be obtained at one time by querying the root instead of flooding those regions, thus incurring significant energy savings. The performance and cost of the algorithm are analyzed and evaluated. The results show the proposed algorithm is efficient and effective in dealing with data aggregation.**

Keywords: **Wireless sensor network, splay Tree, data aggregation, polynomial regression .**

I. INTRODUCTION

Data aggregation is a common operation in sensor networks. Traditionally, information sampled at the sensor nodes needs to be conveyed to a central base station for further processing, analysis, and visualization by the network users. Data aggregation in this context can refer to the computation of statistical means and moments, as well as other cumulative quantities that summarize the data obtained by the network. Such accumulation is important for data analysis and for obtaining a deeper understanding of the signal landscapes observed by the network. The existing researches have analyzed the aggregation algorithm for the application of the sensor network[1],[2],[3].Taking the memory access of aggregation algorithm and other factors into account, an optimized aggregation algorithm[4] was proposed, but there is no consideration of data multiple-hop transmission. Data compression algorithms based on wavelet transforming for sensor networks were proposed in Literature[5]. It can reduce the energy cost of nodes in data transferring efficiently for sensor networks, so, it can prolong the lifetime of the whole networks to a greater degree. But it had not considered the algorithm processing energy consumption and multi-hop path. Literatures [6], [7] consider the energy optimization separately from the angle of the path transmission quality and the path energy consumption to extend the lifetime, but they have not considered the data aggregation. Analysis of data aggregation algorithm indicates that[1] seeking for the optimal aggregation tree on the condition of complete aggregation equates to solving NP-Complete problem of the minimum Steiner tree. According to this NP-Complete problem, literature [8] has considered the balance of the computation processing energy consumption and the transmission energy consumption, and the case without complete aggregation, but it does not involve the overall multi-hop energy consumption and by this constructing aggregation tree. Moreover, considering the computation of measure was done by the sink node. Literature [9] proposed the shortest path tree algorithm, and in this algorithm each source node transmits data along the shortest path to the gathering node. If these paths overlap with each other, and carry on data aggregation alternately in the overlap section, this algorithm is less complex and with less delay of network time. But its energy saving effect was greatly affected by the network topology and cannot win great satisfaction in most cases.

Through constructing the coverage like the Voronoi, and choosing the appropriate quantity and the position to optimize the data aggregation, it may reduce the data quantity transmitting to the Sink node[10],[11],[12]. Literature [10] proposed the algorithm of the greedy aggregation tree. Its shortest path was constructed between the first source node the Sink node arrives and the nearest source node of the tree afterward. However, using this method, Sink can not learn the sensed value but through high cost flooding in the given scope, and once the gradient vector was established, it will not change in the implementation. In LEACH[12], a node set was chosen according to clusters, which clusters each node will join to rely on the node and the clusters communication cost. However, as only a very few nodes act as the role of clusters, from which the appropriate Sink node is far away, clusters will consume excessive energies as a result of the transmission data to the base station. In literature [11], a boundary node possibly belongs to more than one voronoi unit; in this case, if Sink sends out the related data inquiry in the interest region, if necessary, this boundary node must route enquiry request, which will form the bottleneck.

The distributional nucleus regression[13] share similar aspects with this paper's algorithm, but there are great differences. As for the former, every node has its approximate coefficient in its local region scope, thus it cannot reply to the inquiry correctly which involves outside its local region. But in the algorithm of this paper,

* Corresponding author: S.-K Zhang, Tel.:+86 0512 65241247,
E-mail address:zhangsk@suda.edu.cn, Postal address: No 1 Shizhi Street Suzhou China 215006.

doi:10.4304/jcp.5.4.492-499

the coefficient is transmitted upward after the child node is compressed and aggregated. Therefore, the root node of splay tree obtains the final data set of approximate coefficient about its entire covered region. Now, Sink obtains monitor value of any interested region position through the direct inquiry to the root node, and each node sends information containing a vector, which is used to describe the coverage of its local area; the size of this vector increases with massive neighbor nodes which share the nuclear variable along with it. However, in this paper, in view of data aggregation algorithm of event triggering driving based on splay tree, the quantity of the data packet which transmits through each node is constant, and the demand about the node function is simple. It works only if we guarantee that the node can correspond with the constant power in a small scope, and it has certain memory function. There is no need to add the special function node in the sensor network.

II. NETWORK MODEL

In this paper, suppose N static sensor nodes with resources limited randomly are deployed in monitoring area $R = (r \times r)$, denoted by a set $S=(s_1, s_2, \ldots, s_N)$, where s_i represents the first sensor node, as illustrated in Fig. 1. Each node has its location information through triangulation[14], and the location of the sensor, s_i is represented by (x_i, y_i), and each node has its unique *ID*, the same capacity of calculation communication and energy resources. The node achieves the loose time synchronism through Time Synchronization Service[15], the communication access reduces the channel conflict via CSMA/CA. The goal of this paper is to construct the Aggregation Tree (AT) in this N nodes network, where AT is consisted of Nt nodes called Tree Node, which is used to receive and aggregate data, the other $(N - N_t)$ nodes are referred to as non-tree (NT) nodes. Each NT node senses its environmental parameter(s) and reports it to its nearest tree node. The AT is well spread over the entire WSN so that N_t tree nodes are uniformly distributed in the network. In this way, it ensures that the attribute readings sent by NT nodes to the corresponding tree node incur a smaller hop count, and thereby prolongs the overall lifetime of the NT nodes. For simplicity, we use P_{event} (denoted by the dashed rectangle in Fig.1) to represent an event and the event region is denoted by the area, R_{event} where $R_{event} \subseteq R$ .The normal phenomenon is assumed to have already been sensed in the network by the entire AT. R' is defined as the portion of R not occupied by any event, $R' = R - R_{event}$..

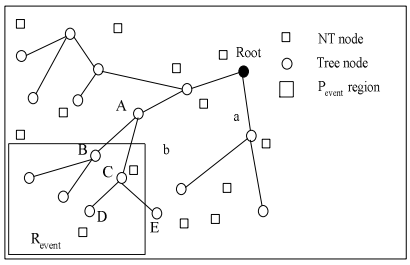

Fig. 1 Network model

III. GENERATION OF AGGREGATION TREE

The algorithm proposed in this paper is that deploying Decide_Root algorithm first to determine some nodes as tree root, then calling *AT_FORM* algorithm to form the splay tree based on the sensor node. Once node receives the information of its child nodes, it will transmit an aggregation package. The root node only transmits the final information including the sensed attribute. In the following part we will give a concrete description of the algorithm.

A. Choose of Root Node

Generally speaking, the occurrence of some events is thought to be the unusual change of the environment condition (e.g. temperature, humidity, pressure and so on), which possibly appears in many ways, such as unusual change of sudden sensed parameter or continual changes with the time passing by. At the same time, many event attributes have the characteristic of time-space correlation and continuous gradual variation in the two-dimensional Euclidean space [1]. With the lapse of time, if the sensor's reading maintains steady, we will suppose that these attributes are the time-space correlation to the sensor monitor, and these sensors have a close relation with readings of other sensors located in the same region. Obviously, the occurrence of some event can trigger partial nodes in the WSN. It might be only one named the isolated spot or many, among which a spot closest related with this event from those triggered nodes is named the event spot, and which be taken as the root to construct aggregation tree expanding to the entire region. Algorithm Decide_Root is to find the event spot C to have the region event selected, and to take it as root node to construct aggregation tree.

Let triggered nodes set consisted of apex set V (G) of diagram G, one-hop neighbor set of the node v is N_v .d-hops neighbor set is N_v^d .Tree generated by N nodes of connected graph G need N-1 sides[4]. It is not difficult to see that the time spent on the aggregation is closely related to the distance from the Sink node to the farthest node. When the region of the event is larger, it will possibly cause a bigger time delay. Let MVV(i) $= \max\{d(i, j)\}$ denoted the maximum distance from i to any ^{j} node. If apex x of graph G, satisfied MVV(x) $= \{\min\{d(i, j)\}$, x is the event spot, i.e. the event spot is the ^{i}nearest apex distant from the maximum apex [4].

After node s_i is triggered, it broadcasts its own id_i immediately. Node s_j which has not been triggered receives id_i from node s_i, and discards it. Triggered node s_k receives id_i from node s_i, and reads it in the buffer, meanwhile renews to hop number of node s_i. If node s_i has not received any node *ID* sending out *ID* broadcast in T_g seconds, then node s_i is the isolated node, and the data transmission starts. Otherwise, when the buffer content of triggered node is non-null, broadcast its buffer content, when node s_i receives a broadcast buffer, add one to hop number with the Different clause id correspondence; if s_i buffer doesn't have this clause id, add one to hop number

of this clause and read into it the buffer; if the buffer has this clause id, and hops value of this clause which add one is bigger than or is equal to hops value of the original clause in the broadcast information, then do not renew this clause; otherwise renew this clause by adding one to hops value of this clause in the broadcast information. Until no more renewals of any node buffer ,then this node stated that it is the event spot, and broadcasts the stop package, whose content includes time label, node id, waiting time threshold T_{wc} and decline parameters of threshold time ΔT and so on. Seeking for event point algorithm *Decide_Root*:

```
1 At the original state the node buffer is null, node id can be
  distinguished in the local range;
2  While (a node s_i be triggered || s_i.time<Timeout) {
3    s_i broadcastingIDbr_i();
4    if (node s_j be triggered and recevingIDbr_i () {
3.   write ID_i to buffer and [hop]i=1; // triggered node s_j receives
     IDbr_i of node s_i, reads id_i in the buffer, hops number from s_j to
     s_i is 1
4.  If (a node s_j not be triggered and receiving IDbr_i) {
5.    dump IDbr_i; // discard received IDbr
6.  }
7.  If (s_i.time>= Timeout) {
8.    s_i is isolated point ;  // s_i is isolated point
9.    data transmitted;
10.  exit();
11. }
12. While(node s_i be triggered and not buffer empty) {
13.   s_i. broadcasting BUbr (); //
14.   If (s_i.receving BUbr() and (!s_i.has(BUbr)) {
15.    s_i.add();  // hops number of this clause adds 1 in the Broadcast
       information
16.  }
17.  Else if(s_i.different()) {
18.    updata s_i; // Renew this node buffer
19.    s_i. broadcasting BUbr();
20.  } Else not updata; // Does not renew this node buffer
21. }
22. End While
23. s_i.say();  // s_i declares that it is the event spot;
24. }
25. End While
26. s_i.broadcasting STOP();//broadcast the stop package, including
    time label, node id, waiting time threshold T_wc and threshold time
    decline parameters ΔT and so on.
```

After the algorithm execution finishes, the event spot *C* of the region event will be found. Before the node broadcasts the buffer content each time, it needs very little time delay T_a. Its purpose is to keep enough time for the node to receive the stop package, and to avoid the phenomenon that many nodes successively state for an event spot.

Suppose that there are *N* nodes in graph *G*, the number of hops from event spot to the farthest apex is *h*. In graph *G* all nodes finishing broadcast one time is called one round. Through broadcasting buffer information, hops number of the farthest neighbor that all nodes can sense adds one at each round process. At the *h* th round, at least one node has already integrated all nodes to its own sensed area in graph *G*, and not renew neighbor scope of these spots at the (*h*+1)th round any longer, by this stage algorithm terminates. Therefore, the number of sending probe packets Probe altogether is:

$$[NUM]_{cen}=N(h+1) \quad (1)$$

B. Construction of Tree

In order to guarantee that AT diffuse to the entire network, the sensor node transmits the sensed value to the corresponding tree node by small hops, holding the topological stability of dispersion node as far as possible, to maintain the original good sensed coverage area, we introduce the graph Voronoi as well as the Delaunay triangle network related to describe the sensor network topology, and based on the definition of Delaunay triangle, splay tree is constructed in Wireless Sensor Networks by taking the central node as the root. The splay tree is one kind of binary tree, and its superiority lies in that it does not need to record the redundant information used in the balanced tree. Let e is a spot of plane, then

$$VR(e)=\{p\in R^n \mid d(p,e)\le d(p,e'),\forall e'\ne e, e'\in E\} \quad (2)$$

is called the polygon Voronoi. Then graph Voronoi is defined as

$$VD(E)=\bigcup_i VR(e_i) \quad (3)$$

i.e. set of all polygons Voronoi in plane, but the triangular Delaunay network is formed by the polygon organic center for connecting all neighboring V-. The triangular net Delaunay has many important properties [14], it can obtain the neighboring nodes' information through the Delaunay expression, Furthermore, and it can be used in searching for the closest node. Based on Delaunay description in the sensor network, we can construct the splay tree taking the node which is definite by the Decide_Root as the root. Let the target sector is A, sensation node collection in the region is

$$S=\{s_i(x_i,y_i) \mid s_i\in A\} \quad (4)$$

Where, (x_i,y_i) is the position coordinate of the known node s_i. In addition, let the weighted graph correspondent by the node collection S network is G in the region, distance of neighbor nodes is the weight of each side corresponding. And Let external memory of the sensed region is in points set $K=\{k_i(x_i,y_i) \mid k_i\notin A\}$, then take node s_i in the target sector as the center regarding the set of points K node extension tree is definite as *T*, has

$$T(s_i->K)=\bigcup_i path(s_i->k_i), k_i\in K \quad (5)$$

Where, $path(s_i->k_i)$ is the greatest span path from node s_i to node k_i in graph *G*, its length is *l* [15]. In this path, the minimum distance between each node is bigger or equal to the minimum distance in any other path from s_i to k_i and the node number is the smallest in graph *G*. The greatest span path had reflected an extension circuit between two nodes. What needs to be pointed out here is that in a specific undirected graph *G*, the greatest span path between two spots is not unique, possibly has multi-strips. But for the different extraterritorial node set, the splay tree of taking the root node as the center corresponding is also not unique.

Let the depth of the tree be p, and the tree node saves the attribute of the same type. Such tree was considered balanced, which reduced data loss and increased accuracy of data aggregation[15]. Algorithm Form_AT constructs a splay tree with given depth and each root node constructs

splay tree of aggregation nodes by running this algorithm. When a node chooses its two children, it will choose the two biggest span nodes, ensuring that the tree covers more sensed regions as far as possible when diffusion. In the process of the multiple regressions, it can achieve the high accuracy, and may reduce the redundancy of the dissemination monitor value. After the splay tree is formed, in each sub-domains all surplus nodes send data to the nearest tree-node away from themselves. In this paper, construct a tree through three kinds of information: Beacon, Probe and JOIN. Fig. 2 described the process about the exchange of different signals to construct the query tree.

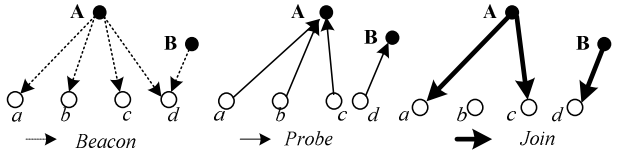

Fig .2 Exchange of signals to construct the aggregation tree.

(1) Beacon Message

In the discovery process of d-hop neighbors, each node *u* broadcasts Beacon news NM_u= {*ADV_discovery*,*u*,*Hop*} in *d*-hop scopes, where, Message is the type of information, hop fields is the count of the information hop, whose initial is 0, and *u* is the node ID. The *v*-node which received NM_u adds u to its own *d*-hop neighbor list N_v^d. For received one or more redundant NM_u, *v* chooses Hop value of the minimum NM_u, its Hop field plus 1 and update $Hop_{u,v}$=*Hop*, if Hop <d continues to broadcast to the neighbors, and then *v* enters waiting state. When V in the waiting state,listening to the $Hop+1<Hop_{u,v}$, the NM_u of *v* ,then record the information and Hop field adds 1, repeat the process; otherwise *v* does not make any actions until the timer time-out. Here the timer Length can be set as long as the network initialization phase. From different nodes of the Beacon, the receiving node v runs the same process. Fig. 3 is the automaton representation of Beacon message process. Through the Beacon news broadcasting each node v can obtain node set in the *d*-hop scope, $N_v^d = \{u \mid Hop_{u,v} \le d\}$ $Hop_{u,v}$ denotes the approximate most short-path hops between the node *v* and *u* which obtains by renewing the broadcast retransmit process with hops. Since the static deployment of sensor networks, for each node u, N_u^d is stable. This N_u^d is set of all possible tree node members. in the layer J , each node broadcasts a Beacon package to all its neighbors of the next hop, thus, the nodes of level j+1 receive many Beacon packages from level j, and stochastically selects a node with probability value p' > p ' 'as their father node, and transmits a Probe package to it.

We can see from Fig. 2, node d receives Beacon from *A* and *B*, *p″* (a input of this algorithm) is optimized by the following method. Set that the father node i broadcasts its Beacon news with correspond radius r to fan-shaped regions with the center of circle angle 120^0C, and this sector's area is $\frac{120}{360}\times\pi\times r^2$, the number of nodes in the region:

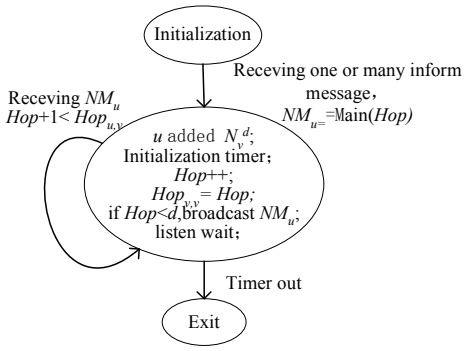

Fig.3.The processing procedure of *Beacon* on v

$$n_r = \frac{120}{360}\times\pi\times r^2\times p \qquad (6)$$

Following the binomial distribution, to choose the region probability p" for the two nodes as their father nodes. $p'' = C_2^{n_r}\times 0.5^2\times(1-105)^{n_r-2}$

If nodes receive a number of Beacons from the expected father node, then their *ID*s will be kept by the parent node of the new choice, in order to resume quickly when the node fails.

(2). Probe Message

The Selected father node waits to receive Probe the package from its child node, afterward, decides which two to take as its child nodes. Once it receives message of all the nodes in its next-hop *j* +1, it will choose two farthest nodes away from it as its children. Those nodes that have not been chosen by any father node will no longer choose to become a member of the AT and transform into the official sensor nodes, in this way, the size of the tree (because the cost of the sensor communications is far greater than of the cost of storage) can be reduced appropriately. As can be seen from fig. 2, node *d* chooses *B* as its parent node, to which it sends a packet probe, then, the node *A* sends beacon to node *a*, *b*, *c* and *d*, but only node *a*, *b* and *c* receive probe packages.

(3). Join Message

Father nodes in the selection of children send them *a* join message, and announces that they will join the tree. Fig. 2 shows, *A* choice of nodes *a* and *c* for their children, and the distance from a and c to A is further than that of either from *a* to *b* or from *b* to *c*. The pseudo-code of algorithm is in the following form.

Construction of splay trees algorithm *Form_AT*(*p*,*p*")

Input: the depth and the b-value.

Output: a binary tree T_c rooted at *r* of depth at most *p* and a unique *ID* assigned to each node of T_c.

1. Begin
2. For each level *j* from 0 to *p*- *l* /* *l* is the largest span path length of inter-node */

   For each node i from *l* to 2^j

3. M_i is a node at level $j + l$
4. n_i is a node at level j
5. n_i sends *Beacon* packet containing n_i's *ID* a_{n_i} to M_i
   Where distance between n_i and M_i < r /* r is the correspondence radius */
6. M_i chooses n_i as its parent with probability > p"
7. M_i sends *Probe* packet to n_i
8. n_i waits *NWAIT* time (which is a sufficiently loog fixed time period) to receive *Probe* packet from each M_i who selected n_i as parent
9. End
10. End

The sensor network, the node in the path of the largest span has good dispersion, reduced the influence of capacity of network-sense due to the overlapping coverage, therefore these dispersive good nodes need to maintain. Through the definition of the splay tree, determined nodes set needs to be maintained in the sensor network.

IV. DATA AGGREGATIONS

The main idea of a tree based data aggregation algorithm is to use the transmission model M being able to fit more monitor data instead of the monitor data of transmission nodes, to reduce the capacity of data transmission, thus saves the energy of the sensor node. Therefore it needs to consider the relations between the cost of the return model and data quantity it may fit. The smaller the cost of transmission model, the more data it can express, the more energy-saving. Because the monitor value of node is often subject to many factors, we expect to fit the most data with the minimum cost mode, and the multiple linear regression models is exactly in line with this goal.

In splay tree each node receives and stores data reported by the recent non-tree node cyclically to it, namely the NT node is responsible for the sensation, but AT node is responsible to store, here, the value saved in AT node is regarded as the function value of the *x*-*y* coordinate. This process describes by three-topple (*f*, *x*, *y*), i.e. *f* is the attribute value transmitted by node located at (*x*, *y*). Data tuple of node *i* stored in AT produces the approach function $f_i(x, y)$, and the progressive function f (*x*, *y*) by the input of the three variables (*z*, *x*, *y*) forms the implementation of multi-polynomial functions, data in such tree node may denotes by multiple regression polynomial function. The following is to discuss the process of carrying out the data aggregation through the polynomial regression on the splay tree.

In general, the form of multi-dimensional linear regression function is as follows [16]:

$$y = f(x_1, x_2, ..., x_m) = a_0 + \sum_{k=1}^{m} a_k x_k \qquad (7)$$

where $x_1, x_2, \ldots, x_m$ is independent variable of the forecast model, y is the sample value with n dimensional vector, which denotes from specific level x_k to estimated the value of node a, and $\vec{a}$ is $(m+1)\times l$ dimensional vector estimated value of a. Using the least square criterion, causes the quadratic difference to be smallest.

$$F(\vec{a}) = (X\vec{a} - \vec{y})^T (X\vec{a} - \vec{y}) \qquad (8)$$

where $$a = [a_1, a_2, \cdots, a_n]^T \qquad (9)$$

The essential condition of existence minimum is the $F(\vec{a})$ partial derivative is zero, then

$\nabla_{\vec{a}} F(\vec{a}) = \nabla_{\vec{a}} (X\vec{a} - \vec{y})^T (X\vec{a} - \vec{y}) = \vec{0}$ again

$$X = \begin{pmatrix} 1 & x_{11} \cdots & x_{m1} \\ \vdots & \ddots & \vdots \\ 1 & x_{1n} \cdots & x_{mn} \end{pmatrix} \qquad \vec{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

$$\nabla_{\vec{a}} (X\vec{a} - \vec{y})^T (X\vec{a} - \vec{y})$$
$$= (\nabla_{\vec{a}} (X\vec{a} - \vec{y}))^T (X\vec{a} - \vec{y}) + (\nabla_{\vec{a}} (X\vec{a} - \vec{y}))^T (X\vec{a} - \vec{y})$$
$$= 2X^T (X\vec{a} - \vec{y}) = 2X^T X\vec{a} - 2X^T \vec{y} = 0 \qquad (10)$$

Then: $$X^T X\vec{a} = X^T \vec{y} \qquad (11)$$

If $X^T X$ is irreversible, there is a solution. In the equation (11) on both sides is multiplied by $(X^T X)^{-1}$, has $\vec{a} = (X^T X)^{-1} X^T \vec{y}$.

Using the polynomial regression, we can obtain the following equation.

$$X = \begin{pmatrix} 1 & y_1 & y_1^2 & x_1 & x_1 y_1 & x_1 y_1^2 & x_1^2 & x_1^2 y_1 & x_1^2 y_1^2 \\ 1 & y_2 & y_2^2 & x_2 & x_2 y_2 & x_2 y_2^2 & x_2^2 & x_2^2 y_2 & x_2^2 y_2^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & y_n & y_n^2 & x_n & x_n y_n & x_n y_n^2 & x_n^2 & x_n^2 y_n & x_n^2 y_n^2 \end{pmatrix}$$

$$Z = \left[z_1, z_2, \cdots, z_n \right]^T, \beta = \left[\beta_1, \beta_2, \cdots, \beta_n \right]^T .$$

where, $$\vec{\beta} = (X^T X)^{-1} X^T \vec{Y} \qquad (12)$$

$$p(x, y) = \beta_0 + \beta_1 y + \beta_2 y^2 + \beta_3 x + \beta_4 xy + \beta_5 xy^2 + \beta_6 x^2 + \beta_7 x^2 y + \beta_8 x^2 y^2 \qquad (13)$$

From the equation (12), we can compute $\vec{\beta}$ with a given location (x, y), and obtain the value of z = p (x, y) is property value of (x, y) nodes. Set $\vec{\beta}$ is $(m+1)\times 1$ - vector, then $X^T X$ certainly is the $m+1$[15] step non-singular. In other words, $n >> m+1$and X cannot denote for weighted linear combination of any other row set. In this paper, the data aggregation algorithm according to the input of the width priority, each tree node has a coefficient from the formula (12) and sends the coefficient set to its parent node. Nodes of each level use the coefficient which obtains from its child to renew sensor attribute value, and these data combine the detection value of node itself to calculat the new coefficient set, and then transfer to a higher level. In the process, to identify the even attribute value in the region is the crux of the matter, because they have a direct bearing on the accuracy of the aggregation, and through the upper and lower bounds the of coordinates of the

{ x_{min} ,y_{min} ,x_{max} ,y_{max} } to identify of the region, where the minimum and maximum value from the son of the father of the current node in the tree under all the sensor nodes. As in fig. 4, set a as the current aggregation node, and data value in the region updates through a. The scope of the region defines by the subtree of *a* to which passed through the smallest and largest coordinates of the sensor nodes. Thus, node *a* gets the border coordinates of the region from its child. Through based on construction of the splay tree and the above description about the process of regression, answer queries every specified time, such as "*SELECT* temperature *FROM* sensors *WHERE* location = (x, y)", or "the highest temperature in target scope "of issues. In the latter case, generate set of (*x*, *y*) coordinates in the designated area, Sink firstly informed of the attribute value of each point location to calculate the maximum value. When Sink needs to know the data of (*x*, *y*), it will send this inquiry to the root, the inquiry by the AT spreading down until the leaf nodes of the last layer.

A data aggregation algorithm based on splay tree *SPAT*(p,n_s) ,n_s is the average number of sensor nodes reporting to the tree node.

1: Begin
2: For each of leaf node i of the tree
file node "*i*" dat is read
multivariate polynomial regression is performed on each data file and the coefficients are stored in the each of the arrays $\beta_0,\beta_1,\ldots,\beta_8$ each of size *N*
End For
3: Initialize level to 2^p
While *p* is greater than 0
sum = level + 2^{p-1} ;*k* = level
While k < sum
4: for each of the non leaf nodes k of the tree computes random x-y points for each of its 2 children *i* and (*i*+l) where (x_{min},y_{min}) and (x_{max},y_{max}) are the coordinates of the leftmost and down most node and rightmost and top most node respectively reporting to the node *i* and *i*+l.
End For
5: Using ($\beta_{i0},\beta_{i1},\ldots,\beta_{i8}$) and ($\beta_{(i+1)0},\beta_{(i+1)1},\ldots,\beta_{(i+1)8}$) new attribute values are calculated and appended to node "*k*".dat.node k then calls the regression function to calculate ($\beta_{k0},\ldots,\beta_{k8}$) and passes it to its parent.
6: End While
7: Level = sum
8: End While
9: End

Compared to the energy consumption and delay of transmission a single data to corresponding location, it is more effective the data is reported by the process of SPAT aggregation. In AT, compression ratio is defined as the byte number sent after the compression of the original data. Set that the depth of *AT* to p and *AT* has a total of $t = 2^{(p+1)} - 1$ leaf nodes, each packet size of w_i-byte includes sensor readings and the location coordinates corresponding to the readings. In this way, the size of byte number which enters to every leaf node is $n_s \times w_i$, where ns is the total number of sensor nodes sent to each tree node. Therefore, byte number transferred to the leaf nodes is $T_l = n_s \times w_i \times 2^p$.

In addition to reading the attribute value from the

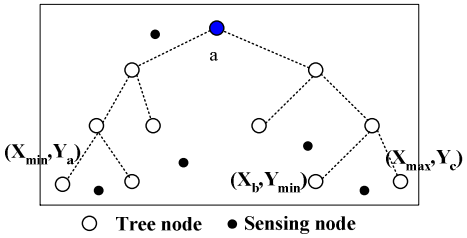

Fig. 4 a node calculate the boundary of the region for data regeneration

sensor nodes, each non-leaf node gets value for input from its two children, to update the coefficient and the *x*-y value of the region border. Therefore, the total byte number inputting to non-leaf node T_{nl} is:

$T_{nl} = n_s \times w_i + 2\times(w_x + w_y + w_c))\times(2^{p-1} - 1 - 2^p)$, get the output packet of byte whose size is $(w_x + w_y + w_c)$ from the tree node (including coefficients and the scope of the *x*-*y*).

The total byte number output from all nodes is $T_0 = (w_x + w_y + w_c)\times(2^{p-1} - 1)$, and then the compression ratio is:

$$C.R = \frac{T_0}{T_l + T_{nl}} = \frac{(w_x + w_y + w_c)\times t}{(n_s \times w_i \times t + 2\times(w_x + w_y + w_c)(t_l - 1))} \quad (14)$$

Where, t is the total number of tree nodes, t_l is the number of leaf nodes.

V. SIMULATION RESULTS AND DISCUSSION

In this paper, discrete event simulation platform *NS*2 was employed to conduct simulation tests; the simulation parameters are shown in table 1 and the focus is to make performance evaluation of the data aggregation algorithm in the following aspects: (1) accuracy of sensor attributes of the whole coverage region, including the absolute value and error percentage, (2)compression ratio, (3) comparison between the sizes of the aggregated packets and non-agrregated packets in the root.

Table 1 Values of simulation parameters used

Parameter	Variation
A	800×800
R	40m
D	1630
A/D	0.0025
A_s	400×400
P ″	0.33
p	4
n_s	12

The definition of the variables was shown in table 1. Supposing the total number of node in region *A* is *D*, then the density of nodes ρ=*D*/*A*, A_s is the sub-region including the single aggregation tree T_c, so the average number *U* of node is determined by A_s in the sub-region.

For complete binary tree, the total number of nodes is t, has:

$$t = 2^{(p+1)} - 1 \quad (15)$$

In addition, n_s set by the front, that is, the average number of sensor nodes reporting to the tree node, in the sub-region the upper bound of node number U[16] is: $U = n_s \times t + t$ or $t = U/(n_s + 1)$, replacement of t using formula (15), results in an optimal solution with the depth of AT: $p = \ln(\frac{U}{n_s + 1} + 1) - 1$,set D=1630, A=800 × 800. $\rho = 1630/800^2 = 0.0025, A_s = 400 \times 400$. Then the average number of nodes in the region is U = 0.002469× 400+ 400=408. Set the depth of the tree p=4, nodes number T_c = t=$2^{(4+1)}$-1=31,n_s=12. ∴ The total number of sensor nodes in this region =31 × 12=372. In fact, the total number of nodes in the region U = 31 +372 = 403 <408. Therefore, the above definition of parameters is effective.

(1).Erro Rate

When the approximation of the actual value, the error rate $E = \left[(|\ z - \overline{z}\ |/z) \times 100\right] \le \varepsilon_{th}$,where ε_{th}=6% is the error threshold, $\overline{z}$, z are respectively the approximate data and the actual data. Fig. 5 has demonstrated the depth of the aggregation tree and relations of the error rate. We can see that with the increase of the aggregation tree depth, the scope of the tree node coverage will be greater; so as to make the whole region can be better monitored, at the same time, the average error and the error rate of similar data fall steadily, that is exactly what we expected. When the depth of the tree is 1, the error rate will reach the maximum, because the tree has only three nodes (in the region the majority of the sensor node dispersed, not in the tree node, and can not be sent to AT) to monitor the scope of the region.

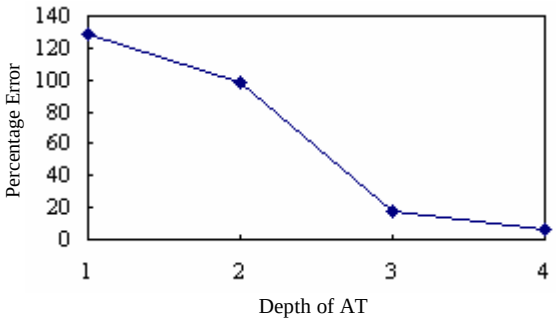

Fig. 5. Variation of percentage error with depth

(2). Compression Ratio

Fig. 6 shows the compression ratio changes with the depth of the tree, as expected, almost constant for 0.02. The decline of the curve shows that with the depth increases, the compression ratio reduces. The deeper the depth of the tree is, the better the degree of compression turns, and the less the output becomes. The high compression ratio reduced the whole information content and thus has saved the correspondence band width and the total energy.

(3). Size of Root Output Packet

Fig. 7 shows the data traffic after data aggregation though *SPAT* algorithm in the root, as well as the root receives data non-aggregation and the normal one-to-many communication. Though the depth of the splay tree is different, when all the nodes of the splay tree implement data compression, the size of packet sent from root to the Sink is a constant, that is, fixed $(w_x + w_y + w_c)$ bytes. The packet size sent from one tree to another tree node by node is almost constant, and it is nothing to do with the size of the network, which makes the total energy of data kept within reasonable bounds. This confirms the above assumptions, that is, each tree node sends the packet which only contains coefficients and *x*-*y* coordinates to its parent node, and the size of the packet is independent of the number of nodes in the tree. In the traditional many-to-one communications, all of the leaves must send data to the root node, so that when the network node increases, the size of data packet transmission growth through the root node with no limits. Through the implementation of this algorithm for data compression, the largest Data communications reduce by the amount of 85% in comparison with [17].

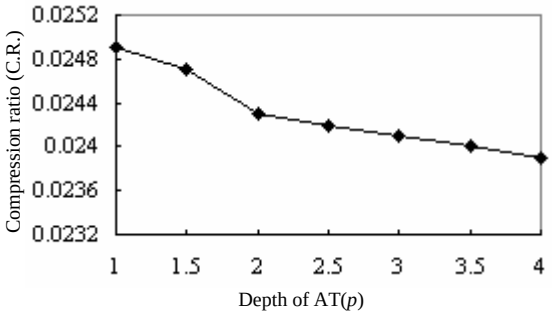

Fig .6. Dependence of compression ratio

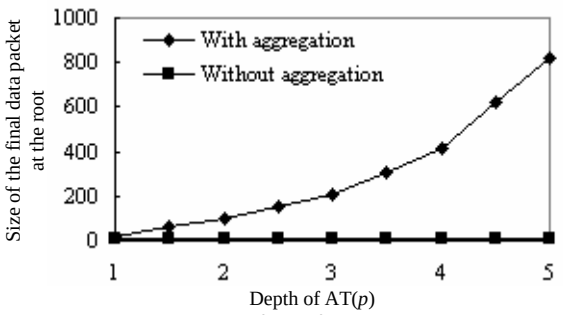

Fig. 7 Dependence of size of data packet (at root node)

VI. CONCLUSION

In this paper, we proposed a novel data aggregation algorithm through the construction of the splay tree, and this algorithm will also be able to detect the event attribute value in the positions where the sensor nodes are lacked. In the construction phase of the tree, the root choice is distributional. It eliminated the request for the overall situation root positional information by sink. By limiting the number of communications, fixed-size information and without taking the depth of the aggregation tree into account, its percentage of error can

be controlled within an acceptable range when data compression ratio remains constant. Simulation results show that the algorithm can effectively improve the perception capacity of the overall network and reduce the energy consumption.

ACKNOWLEDGMENT

Supported by the the National Natural Science oundation of China (60673092,60873116,60873047); Natural Science Foundation of Jiangsu Province of China(BK2008161, BK2008154, BK2009116); The Key Programs of Ministry of Education of China (207040); Jiangsu Provincial Major Program of Science Technique support and independent innovation Foundation (BE2008044); Funded by Preresearch Project of Soochow University; The Opening Project of Jiangsu Province Support Software Engineering R&D Center for Modern Information Technology Application in Enterprise (SX200903); The Higher Education Graduate Research Innovation Program of Jiangsu Province in 2009.

REFERENCES

[1] Li JZ, Li JB, Shi SF. Concepts, issues and advance of sensor networks and data management of sensor networks.Journal of Software,2003,14(10):1717-1727.

[2] Ren FY, Huang HN, Lin C. Wireless sensor networks.Journal of Software,2003,14(7):1282-1291.

[3] Naoto Kimura,Shahram Latifi.A survey on data compression in wirless sensor networks,In proc of the Int'l conf on Information Technology:Coding and Computing.Los CA:2005,2:8-13

[4] Kenneth B,krste A.Energy aware lossless data compression,ACM TransonCoputer Systems,2006,24(3):250-291

[5] XIE Zhi-Jun , WANG Lei , LIN Ya-Ping ,et al.An Algorithm of Data Aggregation Based on Data Compression for Sensor Networks, Journal of Software,2006,17(4):860-867.

[6] Olga Saukh,Pedro Jose,Andreas Lachenmann,et al.Generic routing metric and policies for WSNs,In Proc of 3th European Workshop on wirless sensor networks.Berlin,2006,99-114

[7] Noseong Park,Daeyoung Kim,Yoonmee Doh,et al.an optimal and lightweight routing for minimum energy consumption in wirless sensor networks,In Proc of the 11th IEEE Int'l Conf on Embedded and Real_time Computing Systems and applications.New York,2005,1533-2306

[8] I Kadayif,M Kandemir.Tuning in-sensor data filtering to reduce energy consumption in wirless sensor networks, In Proc of Design,Automation And Test In Europe Conference And Exhibition.Los Alamitos,CA,2004,1530-1591

[9] Dantu,R.AbbasK O'Neill, el al.Data Centric Modeling of Environmental Sensor Networks[C],Global Telecommunications Conference Workshops, GlobeCom Workshops 2004.447-452

[10] Intanagoniwiwat, Estrin, Govindan.Impact of network density on data aggregation in wireless sensor networks,.Proc Of the 22th International Conference on Distributed Computing Systems, July 2002, 575-578.

[11] Henry Dubois Ferriere, Deborah Estrin.Efficient and Practical Query Scoping in Sensor Networks. IEEE International Conference on Mobile Ad-hoc and Sensor Systems, Los Angeles, USA, April 2004.564-566

[12] W. Heinzelman, A. Chandrakasan,H Balakrishnan.Energy-Efficient Communication Protocol for Wireless Microsensor Networks, Proc of the 33th International Conference on System Sciences, Hawaii,January 2000

[13] Guestrin C,Bodik P,Thibaux R,et al.Distributed Regression:an Efficient Framework for Modeling Sensor Network Data,.3th International Symposium on Information Processing in Sensor Networks (IPSN' 04).New York,2004.1-10

[14] Koushanfar F, Potkonjak M, Sangiovanni-Vincentelli A. Error models for light sensors by statistical analysis of raw sensor measurements, In: Proc of the IEEE Sensors, 2004.3:1472-1475.

[15] Ignacio Solis, Katia Obraczka, In-Network Aggregation Trade-offs for Data Collection in Wireless Sensor Networks,INRG Technical Report 102, 2003

[16] Vuran MC,Akan OB,Akyildiz IF. Spatio-Temporal correlation:Theory and application for wireless sensor networks. Computer Networks, 2004,45:245-259.

[17] Tilman Wolf, Sumi Y. Choi.Aggregated Hierarchical Multicast for Active Networks, IEEE Military Communications Conference,2001,2:899- 904.

ZHANG Shu-Kui is currently an associate professor in the Institute of Computer Science and technology at Soochow University, China. His research areas include ad-hoc and wireless sensor networks, mobile computing, distributing computing, intelligent information processing, parallel and distributed systems etc.

CUI Zhi-Ming He is a professor and doctoral supervisor at the Institute of Computer Science and technology, Soochow University and a CCF senior member. His researches areas are intelligent information processing, distributing computing, Deep Web data mining etc.

GONG Sheng-Rong is a professor of Institute of Computer Science and technology in Soochow University, China. He received his the PhD in Computer Science from Beijing University of Aeronautics & Astronautics of China in 2001.His research areas are video processing and communications and computer vision.

LIU Quan is a professor and doctoral supervisor at the Institute of Computer Science and technology, Soochow University and a CCF senior member. His researches areas are intelligent information processing, distributing computing, automated reasoning and GIS etc.

FAN Jian-Xi received the BS, MS, and PhD degrees in computer science from Shandong Normal University, Shandong University,and City University of Hong Kong, China, in 1988, 1991, and 2006, respectively. He is currently a professor of computer science in the School of Computer Science and Technology at Soochow University, China.His research interests include parallel and distributed systems, interconnection architectures, design and analysis of algorithms.