

# Applied Research of Fault-tolerant Monitoring Algorithm in Wireless Networks

Haogui CHEN\*

Hunan City University, Yiyang Hunan 413000, China

**Abstract:** Due to the limited energy of sensor networks, monitoring environmental interference of sensor networks and limited precision sensing components generate the sensing data of sensor nodes with high uncertainty, this paper proposes fault-tolerant monitoring algorithm. Firstly, analyze non-deterministic of sensory data, propose event discovery algorithm of fault-tolerant, and integrate decision-making of events surrounding sensor nodes to determine the probability of occurrence. Secondly, this paper proposes monitoring node selection algorithm, which integrates sensor nodes, event location information and node residual energy, in a given event monitoring quality requirements, select nodes that remaining energy is large and in the event the edge to participate the monitoring tasks. Finally, this paper proposes fault-tolerant event location method, by creating convex hull and estimating the event zone for the event which in the edge. Experimental results show that: the algorithm can greatly reduce the BGP routing convergence time and effectively reduce the BGP routing update message number in the convergence process.

**Keywords:** Network energy; Cluster nodes; Target event; Node moves

## 1. Introduction

In recent years, wireless sensor network monitoring and tracking technology has been widespread concern, there are already some energy efficient event detection methods. Shrivastava N and others proposed monitoring tracking method based on two sensor model, which can effectively reduce network traffic [1]. However, the two sensor models require high reliability of the network, and the uncertainties of network-aware data will affect the tracking results. Zhang W S and put forward a distributed tracking and monitoring method using dynamic tree structure transmitted between nodes collaborate tracking [2]. Zhao F and others put forward the use of information-driven method to determine the participation of the nodes in order to reduce the amount of traffic within the network and node resource consumption [3].

Meng J, and other research the sparse event monitoring technology in wireless sensor networks [4-5]. Considering the number of nodes in sensor network layout and the limitation of network energy consumption, this document uses compressed sensor technology to reduce the number of working nodes, and recommend probability and Bayesian mechanisms to ensure the monitoring probability while reducing the sampling rate. Martincic F and the other put forward distributed event monitoring method, which uses sensor network nodes [6]. Grid mesh cluster head node is responsible for collecting sensory data generated within the node as well as from neighboring clus-

ter's data mean. Mean data in grid in the form of matrix organization, by matrix matching judgment whether the event is detected. Wittenburg G and other design a distributed event monitoring system based on wireless sensor network, be trained which can identify different types of events [7-10]. By making the sensor node detects an event report data, to minimize network communication cost.

The above methods taking fully account into the importance of energy saving within the network, through distributed computing or reduction in working to reduce the number of nodes within the network traffic. However, the data errors and the error data that caused by perception of the network part accuracy and environmental interference is not taken into consideration.

Ding M and the other put forward an event edge monitoring algorithm based on Gaussian mixture model. Based on the correlation of neighbor sensory data model, algorithm uses selection technique divided the nodes within the network into the event edge node and non-edge node these two types. Krishnamachari B, etc proposed the failed node detection algorithm which based on Bayesian theory [11-14]. The algorithm makes data exchange between neighbor nodes to determine the statistical probability of the event. Ding M and other research sensor network node failure events surrounding the identification and monitoring of issues raised locally failed node identification and fault-tolerant edge event monitoring

algorithm, local characteristics of the algorithm make it does not suffer the impact of the network size.

Above methods, use the perceived relevance of data of time or space to find the node data which existed errors, so as to achieve the purpose of fault-tolerant [15]. However, these methods require frequent exchange of information between nodes; it will consume a lot of energy. The proposed method, by quantifying the nodes to offer the likelihood of erroneous data, try to provide a high data reliability node in event monitoring, ensure quality of event monitoring while balancing the network node energy consumption and prolong the lifetime of the entire network .

## 2. Models

For convenience, we make the following assumptions:

a) The sensor network is composed of n same sensor nodes, node set is expressed as  $s = \{s_1, s_2, \dots, s_n\}$ , network model as  $R = \{X, Y\}$ , where  $Y$  is the set of edges of any two nodes. If nodes  $X_i$  and  $X_j$  can communicate, there  $X_{ij} \in Y$ .

b) All nodes within the network can use the GPS or other location technology to calculate the coordinates of your location. With  $(X_i, Y_i)$  represents the coordinates of the node  $X_i$ .

c) The sensing ranges of the network nodes the same as the radius R of the circular area. The communication range of the node and nodes sensing have the same range, or may adjust the communication range of node to sense the range of an integer multiple of the node.

The whole network is divided into a plurality of clusters; each cluster contains a cluster head node and multiple cluster members. The cluster involved in an incident monitoring called working cluster. After the incident, the sensor node of surrounding event generates sensory data, according to the perception of the data to make decisions about whether the event occurred. Nodes in the cluster report the decision-making and the node residual energy information to cluster head node. Cluster head node statistics possibility of occurrence and integrated cluster node energy information, select some nodes in the cluster participate in the monitoring task, the selected node is called the reference node. Cluster head node reports the information of the selected reference to the work of the cluster which has shortest distance with Sink node, then this work cluster locate the event after comprehensive all work clusters data. Because the event occurs only within a local area network, the number of clusters that work at same time is limited, so exchange information between working clusters will not to cause a heavy burden of communication for network.

In order to save network energy consumption, the proposed event monitoring algorithm can be combined with energy management protocol. Most of the existing energy management protocol can be applied to our algorithm.

## 3. Fault-tolerant Monitoring Algorithms

When the target event occurs within the sensor network monitoring area, errors present in the sensing data cannot guarantee that all nodes are able to provide accurate and useful information. That is, some nodes' sensing data cannot be used because the higher the error event monitoring, some nodes of the information provided with redundancy. This section presents fault-tolerant event detection algorithms; the goal is to provide reliable data to select a node in the event location estimation. Taking the needs of the network energy efficiency into account, high node residual energy selected as the reference node can be more effectively extend the network lifetime.

Proposed fault-tolerant event detection algorithm consists of three sub-algorithms, event discovery algorithm, the reference node set selection algorithm and event location algorithms. The table 1 shows the symbols used in the text and their meanings.

Table 1. Symbols and definitions

Symbol	Meaning
K	collection of network sensor nodes $K = \{k_1, k_2, \dots, k_n\}$
T	collection of sensor nodes within the network reference $T = \{T_1, T_2, \dots, T_m\}$
CV	set of the recentest vertexes $CV = \{cv_1, cv_2, \dots, cv_m\}$
$U_i$	Decisions made by node $k_i$ onthe events
$f_i$	the probability of node $k_i$ sending false alarms

### 3.1. Reference node selection algorithm

Firstly, all support the hypothesis  $U_1$  nodes with red, supporting the hypothesis  $H_0$  nodes with blue. For any red node  $X_i$ , denoted by the number of nodes in a red neighbor  $X_i$  percentage of the total number of nodes. If  $<1$ , the node  $X_i$  in the event the edges of the candidate node to the reference node set  $B'$ . The higher the value, the closer the event indicates that the node.

Identifying candidate reference node set  $B'$ , incrementally selected reference node from  $B'$ . In order to use greedy strategy, we define a revenue function g, with, pi, and quantify the node earnings. Greedy-Choice each selection to maximize the gain function g nodes, which try to select the large residual energy, low probability of false alarms and closer node which have shortest distance with events as the reference node. Give set of nodes to support the hypothesis  $U_1$ , assuming that  $B$  is the reference set

of nodes have been selected, and the gain function  $R(X_i)$  of node  $X_i$  is defined as follows:

$$R(X_i) = \frac{k_n^i T_{left}^i}{T_i} \tag{1}$$

According to equation (1) defines greed rules, the paper designs a greedy heuristic algorithm based on Greedy-Choice, as shown in Algorithm 1. Initialization, identifying candidate reference node set  $B'$ , while setting an empty to result set  $B$ . Next, the algorithm selects the current optimal node  $v_i$  into  $B$ . Thereafter, the algorithm enters a loop process, from the collection of the remaining nodes  $B'-B$ , select a node  $v_i$  to maximize the gain function  $g$ , and put the result set  $B$ . When the number of  $B$  is meet nodes required to the event location algorithm requires minimum number of nodes, or nodes of  $B'-B$  is greater than the probability value of false alarms  $K_i$ , exit the loop, the algorithm terminates.

**Algorithm 1 Greedy-Choice reference node selection algorithm**

Input: candidate reference nodes set  $B' = \{B_1, B_2, \dots, B_{v_m}\}$

Output: reference nodes set  $B$

- 1) Blank  $B$  ;
- 2) select a node  $v$  from  $B'$  to join  $B$  to maximize  $R(X_i)$  ;
- 3)  $R(X_i)$  ;
- 4) while (select one node from  $B$  with  $v$  nodes  $\prec$  event location's estimated minimum number of required nodes and  $T_i \succ T_j$  ) ;
- 5) Choose nodes from  $B'-B$  which allow to maximize  $R(X_i)$
- 6) End while
- 7) Return  $B$  .

The algorithm to calculate  $R(X_i)$  takes time  $O(1)$ , so the whole algorithm's time complexity is  $O(k)$ , where  $k$  is the number of nodes in the cluster mean.

**3.2. Event location algorithm**

Because the selected reference node in the event the edge, so the area enclosed by the reference node must contain the incident area. Reference node can form various geometric areas; in which convex hull is smallest convex area including planar the set of points, so convex hull of the reference node is the smallest region that contains events convex domain. In this paper, through the method that establishes the convex hull for reference node to describe the incident area, referred to as H-ES algorithms. H-ES algorithm uses the classic Graham scan method for the convex hull of the reference node. H-ES algorithm is divided into two steps: the reference node sorting and reference node scans. Sorting phase of the reference node in accordance with node x coordinate value in ascending

order, if the node is the same x-coordinate, and then follow the node y coordinate values in ascending order, to get an orderly reference collection of nodes, and express as SC. Reference node scanning phase from the first node within SC, each time you select a node is added to obtain a new convex hull side. Figure 1 is examples that H-ES algorithm describe the incident area. Hypothetical reference number of nodes is  $m$ , then the time complexity of node sorting stage is  $O(m_{log}, m)$ , the time complexity of node scanning phase  $O(m)$ , so the whole algorithm's time complexity is  $O(m_{log}, m)$ .

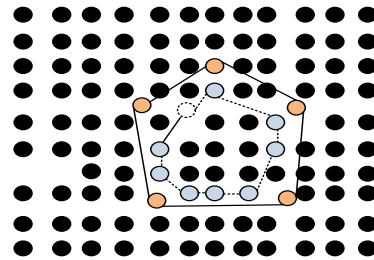


Figure 1. H-ES algorithm diagram

**3.3. Determine the nearest vertex**

From the first node in the set  $C$  begin to identify nearest vertex  $V_1$ . The specific process is:  $V_1$  broadcasts to its neighbors making  $U_1$ , neighbor node receives the broadcast and feedback own decisions to the  $V_1$ . Node  $V_1$  divided neighbor nodes into two subsets based on the contents of the collection: a collection of nodes supported hypothesis  $U_1$  and collection of nodes supported the hypothesis  $H_0$ . Next, calculate the false alarms probability of node that supports the hypothesis  $H_1$ , if the probability is greater than  $K_i$ , it means that the node supports the hypothesis  $U_0$ . through  $Rec1$  central node to construct two perpendicular lines  $l1$  and  $l2$ .  $U_1$  divide the sensing region of  $V_1$  into four sub-areas  $A_1, A_2, A_3$  and  $A_4$ , as shown in Figure 2. Use  $\rho_1, \rho_2, \rho_3$  and  $\rho_4$ , respectively, stands for the ratio between the number of nodes fall and support hypothesis  $U_1$  within each sub-region and total number of nodes fall and support the hypothesis  $U_1$  with  $V_1$  surrounding neighbors of purposes. Suppose the set  $\{\rho_1, \rho_2, \rho_3, \rho_4\}$  is descending order in accordance with the numerical results, according to the different circumstances of four ratio values, determine the closest vertex need to consider the following two situations:

- (1) If  $\rho_1 \neq 1$ , the zone  $A_1$  is the nearest sub-region from the incident region (marked with dotted lines), in this case the vertices of sub-region  $A_1$  is the shortest vertex

distance, as shown in 2 (a) shows,  $A_1 = 5/11, A_2 = 4/11, A_3 = 2/11, A_4 = 0$ .

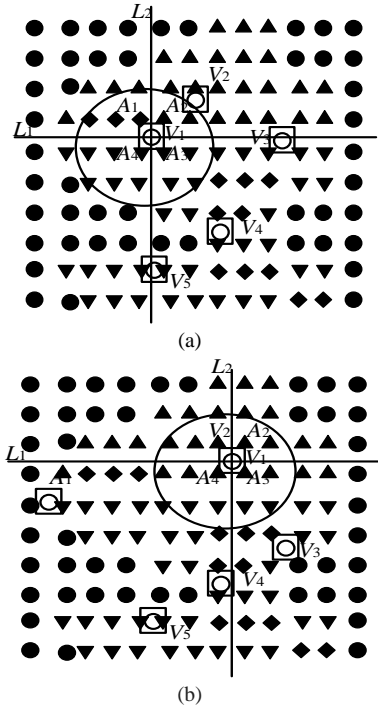


Figure 2. Four sub-areas  $A_1, A_2, A_3$  and  $A_4$

(2) If  $r_1 = r_2$ , and  $r_1 \neq r_3$ , the areas  $A_1$  and  $A_2$  are the nearest sub-region from the incident area (marked with dotted lines), when, the vertex of sub areas  $A_1$  and  $A_2$  at the same distance with event, optionally choose one for nearest vertex, as shown in 2 (b) in Fig.  $A_1 = 5/11, A_3 = 2/11, A_4 = 0$ .

(3) In the same manner, to determine the nearest vertex for the other nodes in the set  $C$ , get the latest set of vertices  $CV = \{cv_1, cv_2, \dots, cv_m\}$ .

### 3.4. Establish security zones convex hull node

Determine the most recent set of vertices  $CV$ , call the H-ES algorithm for establishing the convex hull of the vertices within  $CV$ , to obtain the incident area. Figure 3 shows an example of estimating incident area within the FH-ES algorithm. Algorithm 2 gives the detailed process of FH-ES algorithm.

Algorithm 2, do the executed statement 1-5 takes time  $O(m)$ , do the executed statement 6 takes time  $O(m_{\log}, m)$ , so the whole algorithm's time complexity is  $O(m_{\log}, m)$ .

#### Algorithm 2 FH-ES algorithm

Input: set  $CV = \{cv_1, cv_2, \dots, cv_m\}$

Output: Regional events  $Q$

- 1) for  $i$  in 1 to  $m$
- 2) establish two straight lines  $h_1$  and  $h_2$  which go through the center of  $R_e T_i$  and be perpendicular to  $R_e T_i$ ;
- 3) calculate  $r_1^1, r_2^2, r_3^3$  and  $r_4^4$ ;
- 4) ensure that  $cv_1$  joins  $cv$ ;
- 5) end for
- 6)  $Q = H - ES(CV)$
- 7) return  $Q$ .

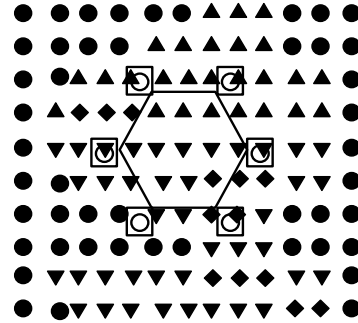


Figure 3. FH-ES algorithm schematic (dashed line indicates the incident event area; the convex hull caused by recent vertices is the result of estimating incident regional)

## 4. Experimental Simulation and Analysis

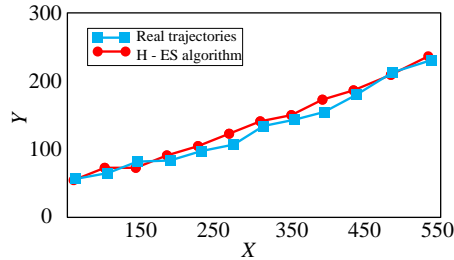
### 4.1. Experimental environment set

In experiments, sensor nodes uniformly distributed in  $800 \times 800$  unit area, the locations of sensor nodes is known. Set all sensor nodes sensing the same range, the initial energy is same. Simulation institute proposes the impact that algorithm works for event monitoring accuracy and network size on event location results. Use the center of event to represent event's location.

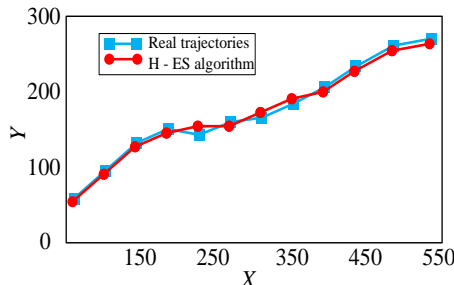
### 4.2. Node monitoring results under resting conditions

The simulation firstly studies the accuracy of two event estimated methods of proposed algorithm. Figure 4 (a), (b), (c) and (d) respectively show the locus of H-ES algorithm trace events center in the case that node within the network under the same. In the experiment, the node sensing radius  $R=30$  units, the number of nodes within the network respectively were 1200, 1600, 2500 and 3600. In the figure, two curves represent the true locomotive trajectory of event center and the trajectory results of H-ES algorithm estimates the event center. X-axis and Y-axis, respectively, stand for the coordinates that event at the location of two-dimensional coordinate system. From the figure we can find that H-ES algorithm can better locate the event. Network size equal to 1200, there are several deviations in trajectory results between H-ES algorithm estimates event and real event. When the network is larger than 1600, H-ES algorithm for tracking is almost not subject to the effect caused by the changeable

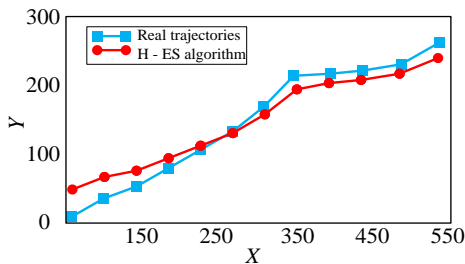
number of nodes within the network. When the network size is 1600, 2500 or 3600, there are two obvious deviations between the estimated results and actual events' trajectory.



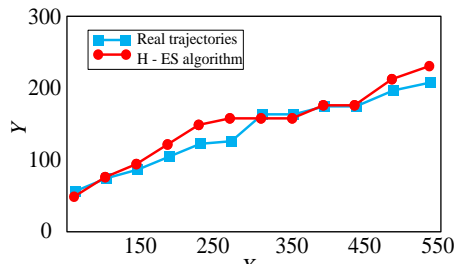
(a) Network size = 1200, real trajectories and H-ES algorithm to estimate the trajectory results



(b) Network size = 1600, real trajectories and H-ES algorithm to estimate the trajectory results



(c) Network size = 2500, real trajectories and H-ES algorithm to estimate the trajectory results



(d) Network size = 3600, real trajectories and H-ES algorithm to estimate the trajectory results

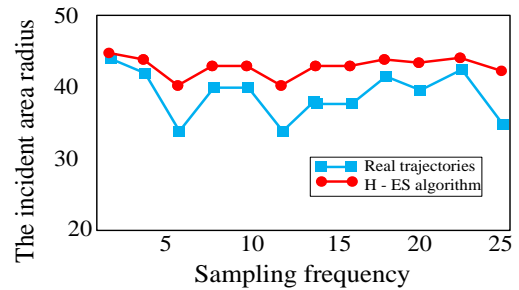
**Figure 4.** (a), (b), (c) and (d) respectively show the locus of H-ES algorithm trace events center in the case that node within the network under the same.

Figure 5 (a), (b), (c) and (d) respectively show the radius' results of H-ES algorithm estimates incident area, at the case that location of the node within the network under the same. In the figure, two curves represent the event

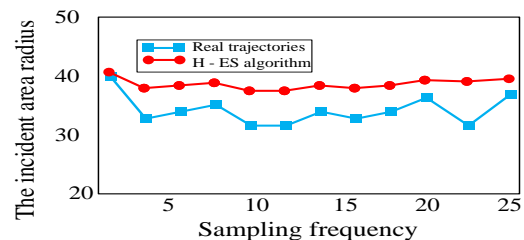
actually happen zone radius values and H-ES algorithm for estimating the value of the radius of the incident area. In the experiment, assuming incidents' region is disc area, H-ES algorithm to calculate the radius of the incident area is the mean that convex hull of the vertices on the distance to the event center. As can be seen from the figure, as the network size increases, the estimated radius value of the incident area tends to be the actual value. The actual radius values of event area fluctuate in value of 30. When the network size is 1200, the maximum difference between real radius value and H-ES algorithm calculate result is 10.4. When the network size is 3600, the difference between these two is reduced to 7.5.

**4.3. The case of node mobility monitoring results**

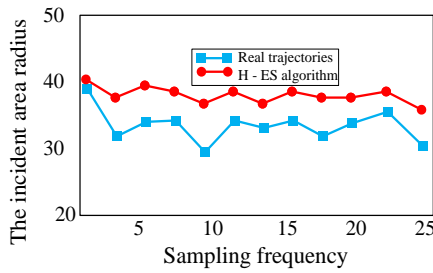
Figure 6 (a), (b), (c) and (d) respectively show the FH-ES algorithm center trajectory tracing event results, at the case that node within the network is mobile. In the experiment, the node sensing radius  $R = 30$  units, the node security zone side length is set to 14 units, the number of nodes within the network were 1200, 1600, 2500 and 3600. In the figure, the two curves represent facts' true movement track and FH-ES algorithm event trajectory estimation results. With the size of the network increases, FH-ES algorithm results improved slightly. When the network is larger than 1200, the reference node nearest vertex area sufficient to produce a convex hull which includes event occurred area, this time, to increase the number of nodes have no significantly affects for estimating incident trajectory results.



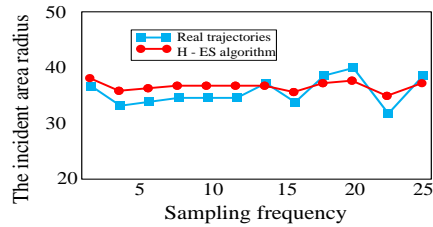
(a) Network size = 1200, H-ES algorithm to estimate the incident area compared with the real value



(b) Network size = 1600, H-ES algorithm to estimate the incident area compared with the real value

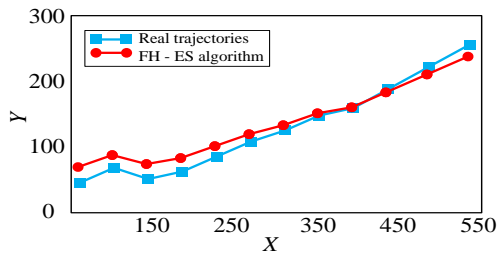


(c) Network size = 2500, H - ES algorithm to estimate the incident area compared with the real value

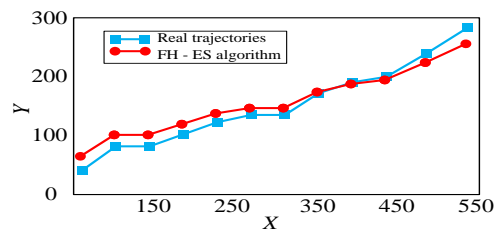


(d) Network size = 3600, H - ES algorithm to estimate the incident area compared with the real value

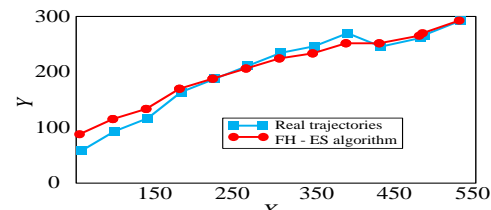
Figure 5. (a), (b), (c) and (d) respectively show the radius' results of H-ES algorithm estimates incident area



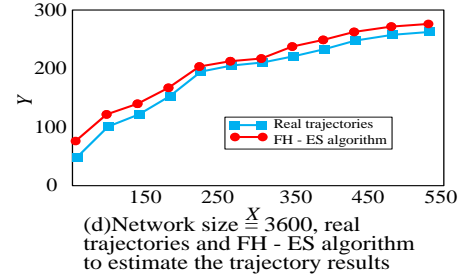
(a) Network size = 1200, real trajectories and FH - ES algorithm to estimate the trajectory results



(b) Network size = 1600, real trajectories and FH - ES algorithm to estimate the trajectory results



(c) Network size = 2500, real trajectories and FH - ES algorithm to estimate the trajectory results



(d) Network size = 3600, real trajectories and FH - ES algorithm to estimate the trajectory results

Figure 6. (a), (b), (c) and (d) respectively show the FH-ES algorithm center trajectory tracing event results

The several groups of experimental results show that when the network size is large enough to increase the size of the network to improve the accuracy of positioning results Event Center had no significant effect. But with the increase in the number of nodes in the network, the gap between estimates the incident area radius and the true value of the radius of the incident area is significantly narrowed. This is because the increase of the network size reduces the distance between nodes within the network. Therefore, the selected reference node distance is closer to the incident area, then estimates closer to the real events area.

### 5. Conclusions

This paper presents a fault-tolerant event detection algorithm. Taking variety of non-deterministic within wireless sensor networks into account, such as sensing data that exists errors and node mobility causes the node position's uncertainty, etc., make relevant fault tolerance mechanisms to reduce the impact that non-deterministic for the quality of monitoring events. Finally, through simulation experiment to propose H-ES algorithm and FH-ES algorithm is correct and effective.

### References

- [1] Abhijit Chandra, Sudipta Chattopadhyay, Role of Mutation Strategies of Differential Evolution Algorithm in Designing Hardware Efficient Multiplier-less Low-pass FIR Filter. Journal of Multimedia, Vol 7, No 5 (2012)
- [2] Andrey N. Pavlov, Sergey A. Terentiev, Vasily V. Kopakov, et al. The System of the Methane Explosion Localization in the Coal Mine Air Trunk. 9th International Workshop and Tutorials, 2008:106-108.
- [3] Karaboga D, Akay B. A modified artificial bee colony (ABC) algorithm for constrained optimization problems. Applied Soft Computing, 2011, 11(3): 3021-303
- [4] Methane Explosion Localization in the Coal Mine Air Trunk. 9th International Workshop and Tutorials, 2008:106-108.
- [5] K.T.V. Grattan, Dr.T. Sum. Fiber optic sensor technology: an overview. Sensors and Actuators. 2000 (82):40-61.
- [6] M. Aykut Yigitel, Ozlem Durmaz Incel, Cem Ersoy. QoS-aware MAC Protocols for Wireless Sensor Networks: A Survey. Computer Networks, 2011, 55(8):1982-2004.

- 
- [7] HaoZhengjie, Zhang Jilong, Frame synchronization technology based on perfect punctured binary sequence pairs, Reviews on Advanced Materials Science 2011.
- [8] Karaboga D, Akay B. A survey: algorithms simulating bee swarm intelligence. Artificial Intelligence Review, 2009, 31 (1-4): 61-85
- [9] Bitam S, Batouche M, Talbi E. A survey on bee colony algorithms. Proc of IEEE International Symposium on Parallel and Distributed Processing, Workshops and Phd Forum: IEEE Service Center, 2010: 1-8
- [10] Karaboga D, Ozturk C. A novel clustering approach: artificial bee colony (ABC) algorithm. Applied Soft Computing, 2011, 11(1): 652-657