International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Predictive Algorithm for Critical Event Management in Wireless Sensor Network

Sulochana M. Gore[1], Dr. Sulochana B. Sonkamble[2]

[1]Computer Engineering Department, Rajarshi Shahu School of Engineering and Research, JSPM NTC, Narhe, University of Pune, Pune, Maharashtra, India

[2]Computer Engineering Department, Rajarshi Shahu School of Engineering and Research, JSPM NTC, Narhe, University of Pune, Pune, Maharashtra, India

Abstract: ***Wireless Sensor Network (WSN) is highly demanded in the field of networking. This popularity of WSN is because of its ability to yield the real time sensory data. This sensory data is further processed to generate useful and sensible results required for an application. The natural disaster detection and management is the main focus behind this proposed project. Natural critical event contributes a large penalty in terms of big loss of living and nonliving asset. In this paper, we proposing an algorithm that, (1) detects the critical event under WSN, (2) calculates direction of growth and speed of the critical event, (3) predicts the next affecting area within anticipated time period. (4) Alerts the prevention system around the WSN. Wireless sensor network is generating the real time data such as temperature, pressure, ambient light, humidity etc. We use this data as an input to the system. The power of computation is applied on the sensory data to make the system functioning as per desire. In this paper, we proposed a predictive algorithm for a critical event detection and management. A prevention system is a set of preventing objects. The algorithm predicts critical event probable spread area and accordingly give alert to all preventing objects. These preventing objects are responsible to take a contrary action against calamity. WSN is programmed to give a activation signal to all those prevention systems or objects which are under critical event probabilistic area. In this way, we can manage the critical event and reducing amount of losses considerably.***

Keywords: Wireless sensor network, predictive system, critical event management, probability distribution, Hidden Markov Model

1. Introduction

The natural disasters like wildfire, flooding, earthquake etc. are the calamity that induces lot of unmeasurable losses such as death, injuries, destruction of assets and many more. The ideation is to reduce the losses, by taking an preventing step against the calamity by predicting the growth and speed of the calamity. The idea is motivated by [2], a health care application which is developed for elderly people, those need a continuous health monitoring system. Such system provides serenity to a dispersed user, that they are under continuous medical observation and immediate medical service is available for them in any critical situation. They implemented intelligent forwarders that provide the wrist wearable remote sensors with context awareness. They transmit only critical information to the big data server for analytics when certain unpleasant behaviors happen to user. This existing system, forwarding the critical information to the preventing organization only when the dispersed user allows it. This require a user permission and evidently human interaction does not allow the fully automation of the system. In our proposal, prevention system is activated automatically to resist the critical event. The probability distribution is used to predict the future growth of the critical event in wireless sensor network. The WSN is programmed to predict all preventing object and initiates those objects against the calamity. The existing system does not estimate the probable preventing systems in wide affected area. It is tightly bounded with the specific preventing organization. The proposed sensor network is resisting the critical event and automatically initiates the preventing object without any human intervention.

2. Problem Definition

2.1 Problem Definition

1) Design: To design energy efficient structure of cluster in wireless sensor network.
2) Monitoring and Filtration: To collect sensory data from WSN and detect critical event.
3) Calculation: To calculate direction of growth and speed of the critical event.
4) Prediction: To predict the expected affecting area using probability distribution.
5) Searching and activation: To search all preventing systems in probable affecting area and alerting them.

2.2 Objectives and Motivations

The primary object of the paper is to tackle with the natural disaster. The natural disaster is uncontrollable event that has lot of negative consequences. This algorithmic approach to detect and take preventing action against such events will reduce losses considerably. This idea is also applicable to manly driven accidents. The intelligence is provided to wireless sensor network such that it calculates the probability distribution of critical event and accordingly activate already mounted prevention system.

3. Literature Survey

3.1 Hotspot Locating attack and Adversary Model [1]

A hotspot is the small region with high volume signals generating in wireless sensor network. This is a noticeable discrepancy in the network traffic for prolonged time period. A Hotspot-Locating Attack is the searching technique of

Licensed Under Creative Commons Attribution CC BY

hotspot attack. In [1] authors have designed an adversary model to save pandas from hunter. This model gives the fake information to the network observing hunters. In the initial phase, the adversary deploys a monitoring device near of the Sink and deploys the other devices at initial observation points distributed in the network. For the monitoring phase, the monitoring devices collect traffic information which includes the following tuple <Pi, Xi, Yi, ti>, where Pi is the content of a packet, (Xi,Yi) is the coordinates of the sensor node that sent the packet, and ti is the time of sending the packet. For the analysis phase, the adversary uses traffic analysis techniques to analyze the collected data to decide to 1) search an area for pandas; or 2) change the locations of the monitoring devices.

The adversary is a hunter who listens in on the wireless transmission network and tries to make use of the network traffic to determine the locations of pandas to chase them. The adversary allocates a group of monitoring devices in areas of interest, called observation points, to collect the traffic information in these areas, but he cannot monitor the traffic of the entire network.

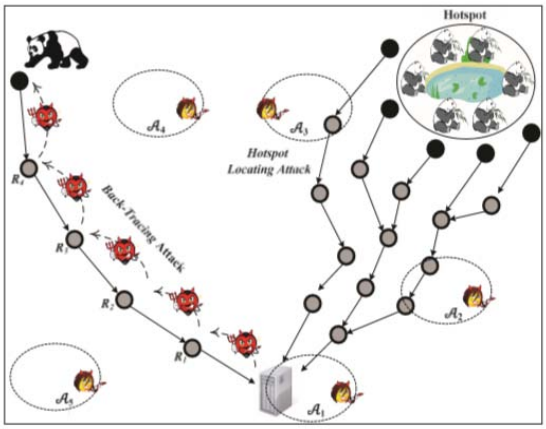


Figure 1: Adversary Model

The adversary analyzes the information collected by the monitoring devices to locate pandas or change the observation points, e.g., to be closer to pandas. For example, Fig. 1 shows that the adversary distributes five monitoring devices in five observation A1to A5.

3.2 Hidden Markov Model (HMM)

Intelligent Information forwarder proposed in [2] uses Hidden Markov Model to continuously monitor the health condition of a patient and makes context awareness about the observed and probable health condition. A wrist wearable device has been designed that includes different sensors for different human body condition measuring parameters like skin temperature and heart beats etc. The integration of these sensors on a chip is facilitated with Bluetooth Low Energy (BLE) technology for connectivity of the wrist device with Android phone of user. Whenever any health anomaly is detected the wrist device establishes human-mobile interaction. If user really needs a health care attention an emergency service is called from preventing organization like hospital where the user is already registered his healthcare. A big data system at service station side is communicating with the user's Android mobile phone through Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) port. The system is designed especially for elderly people who need continuous healthcare monitoring and may require an emergency service anytime anywhere.

Hidden Markov Model generates the hidden probable states by using the visible observed states. The HMM differentiate these states as transition probability and observational probability states. The HMM is exploited in a wide range of applications. In the biomedical science field, for example, the model is ideal for gene-prediction and virus infection outbreak etc. We have taken a virus epidemic model as a case study for our project. With transition probability and observational probabilities, it is possible to identify the most probable state at a specific time step based on the observations made at that point along with the preceding states.

3.3 Virus outbreak probability

The concept of critical event spread is much analogous to virus outbreak epidemics. Any natural critical event such as fire, flood etc. spreads in the area of contact like a virus spread. Once the virus is detected in a person, the particular area around him is probable area of virus spreading. In our literature survey we have studied the epidemic probability distribution model that estimates the next probable area of virus infection. This model uses Approximate Bayesian computation for determining virus infection rate and recovery. As diseases spread causes more deaths throughout the world than wars and famines, so this spirited their study since long time. Over the years, many different models have been discovered and analyzed. Using mathematical models to describe epidemics can be useful in giving estimates for the level of vaccination required. Similarly, the estimation of most probable affecting area is useful for activating the intelligent prevention system in the network to resist the calamity.

4. Existing system

4.1 Event Detection

A sensor node continuously compares the sensor reading with some predefined value to detect the anomaly. When the detected, it reports the event to the sink node. In [1], author has proposed a habitat monitoring application where the WSN is deployed in a wild area for monitoring pandas. They also proposed a design that gives fake information about the panda's location that diverts the hunters who are monitoring the network. Fig. 2 shows the flowchart of a hotspot-locating attack using the adversary model discussed in section 3.1. In the initial phase, the adversary deploys a monitoring device near of the sink and deploys the other devices at initial observation points distributed in the network For the monitoring phase, the monitoring devices collect traffic information which includes the following tuple <pi, xi, yi,ti>, where pi is the content of a packet, (xi,yi) is the coordinates of the sensor node that sent the packet, and ti is the time of sending the packet. For the analysis phase, the adversary uses

Licensed Under Creative Commons Attribution CC BY

traffic analysis techniques to analyze the collected data to decide to whether search an area for pandas or to change the locations of the monitoring device to save pandas by giving fake information

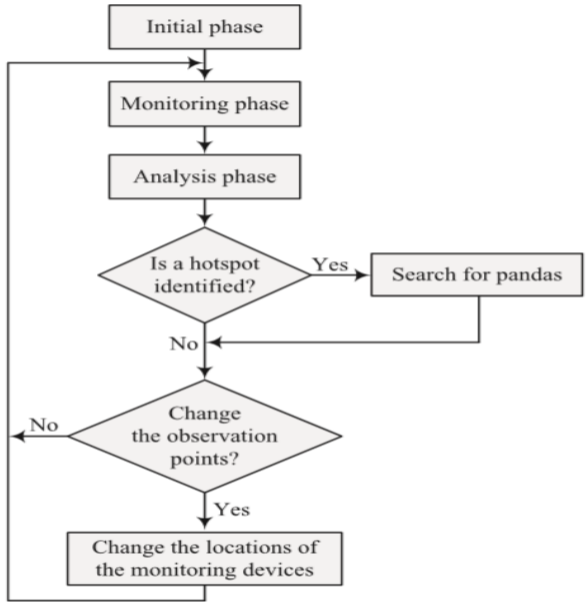


Figure 2: Flowchart of Hotspot locating attack

4.2. Probability Estimation using HMM

The Hidden Markov Model as described in section 3.2 gives the transition probability and observational probability states. As shown in figure.3 there are N hidden states as set S = [S_1, S_2, . . .,S_N], and has M observations from sensors indicating by the set O_t = [O_1, O_2, . . .,O_M], where t =1, . . ., T, and a_{ij} denotes the transition probability defined as

$$a_{ij} = P(q_{t+1} = S_i \mid q_t = S_i), \quad (1)$$

and the observation probability $b_j(O_t)$ in the sensor state j, is given by

$$b_j(O_t) = P(O_t \mid q_t = S_j). \quad (2)$$

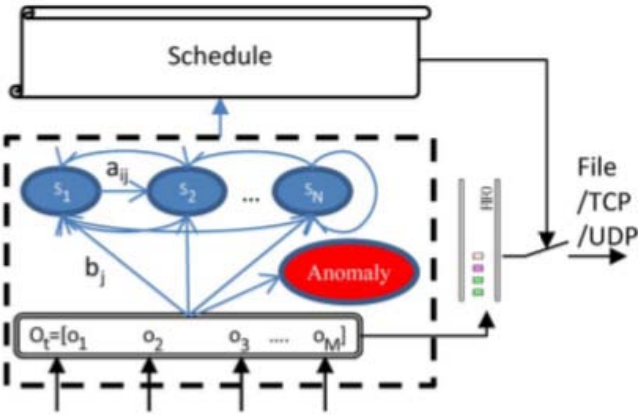


Figure 3: State driven Information Forwarder

The observation sequence O = [O_1, O_2, . . ., O_T] and a model λ = (a_{ij}, b_j, π_j), where i, j = 1, . . ., N, and π_j is the initial probability of state j, the probability of the optimal state sequence $Q^* = q_1^*, q_2^*, \ldots, q_T^*$ can be obtained by Viterbi algorithm [12].

Define

$$\delta_t(i) = \max_{q_1,q_2,\ldots,q_{t-1}} P[q_1, q_2, \ldots, q_{t-1}, qt = Si, O_1, O_2, \ldots, O_t \mid \lambda] \quad (3)$$

where $\delta_t(i)$ is the highest probability along a single state sequence as calculated at time t, accounting for the first t observations and terminating with state Si . The state sequence itself is given in array ψ, which is populated with the state maximizing that probability calculated by δ t at each step.

1) Initialize:

$$\delta_1(i) = \pi_i b_i(O_1), \quad 1 \le i \le N$$
$$\psi_1(i) = 0, \quad 1 \le i \le N. \quad (4)$$

2) Recursion Step:

$$\delta_t(j) = \max [\delta_{t-1}(i) a_{ij}] \, b_j(O_t) \quad 1 \le i \le N$$
$$\psi_t(j) = \text{argmax} [\delta_{t-1}(i) a_{ij}] \quad 1 \le i \le N$$
$$2 \le t \le T \; ; \; 1 \le i \le N. \quad (5)$$

3) Terminate:

$$P^* = \max_{1 \le i \le N} [\delta_T(i)]$$
$$q_T^* = \text{argmax}_{1 \le i \le N} [\delta_T(i)] \; . \quad (6)$$

4) The backtracking procedure:

$$q_t^* = \psi_{t+1} q^*_{t+1}, \; t = T-1, T-2, \ldots, 1. \quad (7)$$

The resulting state sequence ψ is the most possible sequence that has produced the observation at time T, given transitions from previous states.

5. Hypothesis

Before proceeding further, we assume (1) The WSN has divided into sub networks such that optimal number of clusters has been formed. (2) Each cluster has a centroid and each centroid of a cluster can communicate with the nearby cluster's centroids. (3) All sensor nodes in a cluster including centroid can communicate with each other. (4) Prevention system like robot is already mounted around the WSN.

6. Methodology

Energy efficient cluster design is the most prominent way to enhance the limited life of batteries equipped in the sensor nodes. In a sensor cluster, the centroid node is considered as active node and all other sensor nodes are passive or sleeping nodes. Whenever a critical event is detected by the active node, it sends a signal to all passive nodes in a cluster, and then all passive nodes become active and starts searching the critical event. This design of cluster leads the energy efficient design of cluster as shown in fig(5).

We divide the sensors into two groups depending on their functionality in the WSN as:

1) Strategic sensors or Reporters (Active, Passive, Critical, and Recovered) and,
2) Action taking sensors against calamities. (Sink Node)

Licensed Under Creative Commons Attribution CC BY

3) Strategic sensors are active sensors initially. When any calamity detected by this active sensor, it sends a 'wake up call' to all other sleeping nodes.

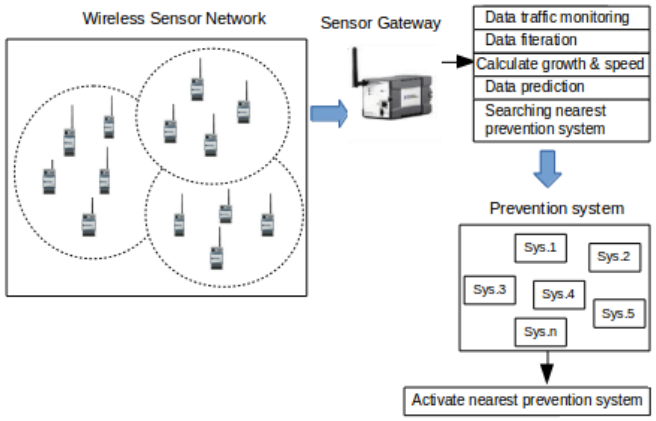


Figure 4: Proposed system architecture

After all sleeping or passive nodes becomes active and starts detecting critical event. There are four observational states of the sensor nodes {Active, Passive, Critical, Recovered}. When active node detects critical event, it changes its state to 'Critical state'. All critical nodes are responsible to send a four tuple <nodeid,xi, yi,ts> information to sink node, where *nodeid* is the unique node identifier, <xi, yi> are x and y coordinates of node i and *ts* is the timestamp when the sensor node i detected the critical event. This information is used to determine the direction of growth of the critical event and it's speed. The average difference between the timestamp parameter of all critical nodes are used to determine speed of the growth. The sink node maintains the *graph* data structure to establish the critical node map to identify growth of the critical event. The speed of a critical event can be calculated as :

$$ts_{avg} = \frac{\sum_{i=1}^{n} (ts_{i+1} - ts_{i})}{n} \quad (8)$$

where, n is the total number of critical nodes and ts_i, ts_{i+1} are the timestamp reported by i^{th} and $i+1^{st}$ critical nodes.

Sink node is also responsible for identifying the probable affecting area by identified critical event. By using this information sink node give an alert signal to all those preventing objects which are present under the probabilistic area of critical event. Fig. (4)depicts the architecture of proposed system. Sink node is responsible for calculation of speed and direction of growth of critical event, Data prediction and searching prevention system. Data traffic monitoring and filtration is done at strategic sensor node which are active or critical.

7. Proposed System Modeling

7.1 Problem Modeling

The Passive -Active–Critical-Recovered (PACR) model is the probability distribution model where we estimate the outbreak of critical event. A sensor node possess either of one state among {Passive,Active,Critical,Recovered }.At starting only one node in a cluster is in active state and all other nodes are in passive state, this is because we have limited batteries in sensor node . As soon as active node found any critical event it sends 'wake-up' call to all other passive nodes in the cluster. Then all passive nodes become active and start searching for critical event. All nodes which found critical event are considered as critical node.

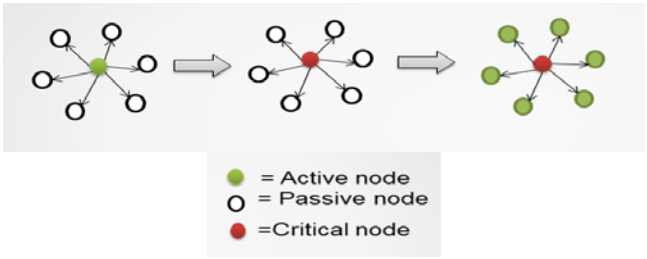


Figure 5: Critical event outbreak through energy efficient cluster organization

All critical nodes are responsible to send information to sink node. Further computations are performed on sink node. If critical nodes are recovered from the critical event they are treated as in the *recovered* state otherwise they are found to be *died* after particular time period.

7.2 The Markovian Probabilistic Model

A process is Markovian if the conditional probability distribution of the future states of the process depends only on the current state of the process and not on any past states, that's why Markovian process is also known as memory less process.

The stochastic process is a family of random variable { X(t) | t ∈ T } defined on common sample space S and indexed by the parameter t, which varies on index set T .The values assumed by the random variable are called *states. Set of discrete state space is I = {P, A, C, R}*

The outcome $\{a_1, a_2, a_3, ..., a_m\}$ belongs to the state space. Markov property states that,

$$P(X_n=i_n \mid X_{n-1}=i_{n-1}) \quad (9)$$

The system is said to be in a_i state at time n.

7.3 Probability Transition Matrix

The transition probability p_{ij} of a cluster from one state i to state j is given by probability transition matrix P,

$$P = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ . \\ . \\ a_m \end{array} \begin{array}{c} \begin{array}{cccccc} a_1 & a_2 & a_3 & ..... & a_m \end{array} \\ \begin{pmatrix} p_{11} & p_{12} & p_{13} & ..... & p_{1m} \\ p_{21} & p_{22} & p_{23} & ..... & p_{2m} \\ p_{31} & p_{32} & p_{33} & ..... & p_{3m} \\ . & . & . & ..... & . \\ . & . & . & ..... & . \\ p_{m1} & p_{m2} & p_{m3} & ..... & p_{mn} \end{pmatrix} \end{array}$$

The probability that the Markov Chain will move from state i to j in n steps is given by,

$p^{(0)}_{ij} = (X_{m+n}=j \mid X_m=i)$

Evaluation of n-step Transition Probability Matrix can be done by using Chapman Kolmogorov equation as:

Licensed Under Creative Commons Attribution CC BY

$p_{ij}(m+n) = \sum_k p_{ik}(m) p_{kj}(n)$

The Probability distribution of the n-step system can be calculated by using initial probability distribution and power of transition matrix, is given as

$p^{(n)} = p^{(0)} P^{(n)}$ (10)

where $p^{(0)}$ is initial probability distribution and P is probability transition matrix.

Fetching the data from critical sensing nodes dynamically to sink node can be predicted using probability distribution transform matrix. Initial cluster state can be passive, active or critical. Each state transformation of a cluster can be given by right stochastic matrix. A stochastic matrix is a matrix that can be used to describe transition of Markov chain. Each entry is representing probability. In right stochastic matrix each row summing to 1. As state of next node is dependent on the state of current state only, because when any critical event detected at current node, then and then only the probability of next critical event sensing node comes into picture. Such chains are called Markov chains. Initial state of a node is active that's why we represent this as a single row matrix [0 1 0 0] as per PACR model. By using the equation (10), the probability of current node after a time unit is given by:

f (n) = [P A C R] * [Transition Matrix] (11)

Initial Condition for PACR is [0 1 0 0] because when any sensor node becomes active then only the process is going to start. The state transition diagram can be given as:

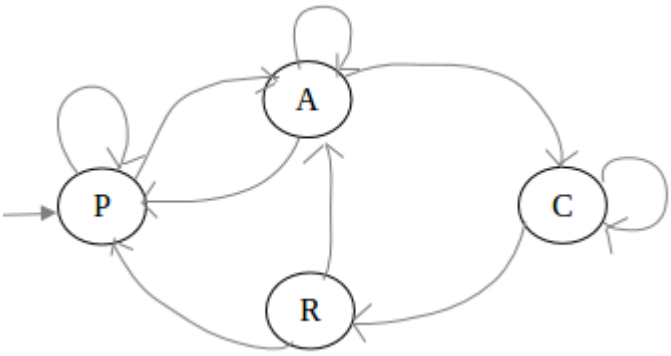


Figure 6: State transition diagram

The computation can be performed by using right stochastic matrix as follow.

$$f(0) = [0\ 1\ 0\ 0] \begin{pmatrix} 1/2 & 1/2 & 0 & 0 \\ 1/3 & 1/3 & 1/3 & 0 \\ 0 & 0 & 1/2 & 1/2 \\ 1/2 & 1/2 & 0 & 0 \end{pmatrix}$$

$$f(0) = [1/3\ \ 1/3\ \ 1/3\ \ 0]$$

$$f(1) = [1/3\ 1/3\ 1/3\ 0] \begin{pmatrix} 1/2 & 1/2 & 0 & 0 \\ 1/3 & 1/3 & 1/3 & 0 \\ 0 & 0 & 1/2 & 1/2 \\ 1/2 & 1/2 & 0 & 0 \end{pmatrix}$$

This matrix multiplication gives the probability of a sensor node after a unit time. As a critical node is responsible to make its in-range sensor node active .We use divide and conquer method. The first critical node state can be calculated using above calculation. We use recursion method as time progresses

f(n) = [0 1 0 0] t_n for n=0
= f (n-1). t_n otherwise (12)
Hence f(n) gives the probability distribution function of an critical event over the time n.

8. Proposed Algorithms

1. Data Filtration: Detection of critical event by active sensor using threshold values. This task is done by active sensor node. Suppose a an active sensor node i is reading a set of multiple sensing parameters like temperature, humidity, light etc. and that are indicated by $R_{it}=\{R_1,R_2,R_3,\ldots R_n\}$. The sensor nodes are programmed with a short program that detects the critical event by providing a threshold value th_1, th_2, th_3, …, th_n to $R_1,R_2,R_3,\ldots R_n$ respectively. If the condition $((R_1>th_1) \wedge (R_2> th_2) \wedge (R_3>th_3) \wedge \ldots, (R_n> th_n))$ satisfy then sensor node becomes critical node.
2. Send Signal: Critical node will send 'wake up' call to all 'in range' passive sensor nodes and send own (nodeid, t) pair to sink node, where *t* is the timestamp when critical event was detected by sensor node with nodeid *nodeid.*
3. Searching critical event: All waked up sensor nodes starts detecting critical event repeats step 1 and 2 by comparing sensor reading with some predefined values.
4. Reporting: All nodes those detected critical event will report to sink node along with <nodeid,xi, yi,ts> as explained in section 6.
5. Computation: Sink Node calculates growth of critical event and speed of critical event.

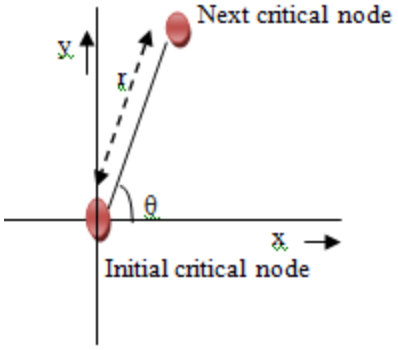


Figure 7: Establishment of node location in graph

Licensed Under Creative Commons Attribution CC BY

Sink node uses directional graph to draw the map of critical nodes. At sink, sequence of operations is performed for evaluating direction of growth and speed of the critical event.
a)Next, as each critical node is bounded with the timestamp, we can calculate the average speed of critical event by subtracting two consequent critical node's timestamp using equation (8).

6. Compute probability distribution using HMM: Depends on parameter calculated in step 5, sink node will identify probability distribution by using HMM, probability transformation matrix and Markov chain.
7. Recursion: Use recursion until all nodes are recovered and we have PACR status as [0 0 0 1] for each cluster using equation (12).
8. Search and activate prevention system: Identify all preventing objects in probable affecting area and give activation alert to all.

9. Conclusion

The focus of the work is environmental critical event detection. Sensor nodes are reporters and responsible to sense environmental dimensions for any critical anomaly detection .The sink node is the actor node and responsible to take the appropriate action against the calamity. This model leads the automation and gives an intelligent system that prevents the more losses due to the natural disaster. Wireless Sensor Networking permits more connectivity for sensor applications and provides advanced control over monitoring, automation for a range of industries. The applications of Wireless Sensor Networks are almost titanic with many industries and applications having specific technology requirements such as reliability, battery life, range, frequencies, and topologies, size of the network, sampling rate and sensor use.

References

[1] Mohamed M.E.A. Mahmoud and Xuemin (Sherman) Shen, Fellow, IEEE "A Cloud-Based Scheme for Protecting Source-Location Privacy against Hotspot-Locating Attack in Wireless Sensor Networks" IEEE transactions on parallel and distributed systems, vol. 23, no. 10,October 2012.
[2] Ping Jiang, Jonathan Winkley, Can Zhao, Robert Munnoch, Geyong Min, and Laurence T. Yang, Member, IEEE "An Intelligent Information Forwarder for Healthcare Big Data Systems With Distributed Wearable Sensors" "IEEE systems journal 1932-8184 © 2014 IEEE.
[3] Daisuke Takaishi, Hiroki Nishiyama, Nei Kato, and Ryu Miura School of Information Sciences, Tohoku University, Sendai, Japan "Towards Energy Efficient Big Data Gathering in Densely Distributed Sensor Networks" DOI10.1109/TETC.2014.2318177, IEEE Transactions on Emerging Topics in Computing.
[4] Chunsheng Zhu, Student Member, IEEE, Hai Wang, Student Member, IEEE, Xiulong Liu, Lei Shu, Member, IEEE,Laurence T. Yang, Member, IEEE, and Victor C. M. Leung, Fellow, IEEE "A Novel Sensory Data Processing Framework to Integrate Sensor Networks With Mobile Cloud"IEEE systems journal 1932-8184 © 2014 IEEE
[5] Chunsheng Zhu, Hasen Nicanfar, Victor C. M. Leung, Wenxiang Li, Laurence T. Yang Department of Electrical and Computer Engineering, The University of British Columbia, Canada. "Ad-hoc and Sensor Networking Symposium A Trust and Reputation Management System for Cloud and Sensor Networks Integration" IEEE ICC 2014
[6] P. Indyk and R. Motwani, "Approximate nearest neighbors: Towards re-moving the curse of dimensionality," in Proc. 30th Annu. ACM Symp.Theory Comput., Dallas, TX, USA, 1998, pp. 604–613.
[7] M. Slaney and M. Casey, "Locality-sensitive hashing for finding nearest neighbors," IEEE Signal Process. Mag., vol. 25, no. 2, pp. 128–131, Mar. 2008.
[8] P. Indyk and R. Motwani, "Approximate nearest neighbors: Towards removing the curse of dimensionality," in Proc. 30th Annu. ACM Symp.Theory Comput., Dallas, TX, USA, 1998, pp. 604–613.
[9] M. Slaney and M. Casey, "Locality-sensitive hashing for finding nearest neighbors," IEEE Signal Process. Mag., vol. 25, no. 2, pp. 128–131,Mar. 2008.
[10] D. W. Aha, D. Kibler, and M. K. Albert, "Instance-based learning algorithms," Mach. Learn., vol. 6, no. 1, pp. 37–66, Jan. 1991.
[11] T. Cover and P. Hart, "Nearest neighbor pattern classification," IEEE Trans. Inf. Theory, vol. IT-13, no. 1, pp. 21–27, Jan. 1967.
[12] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," IEEE Trans. Inf. Theory, vol. IT-13, no. 2, pp. 260–269, Apr. 1967.
[13] Using Probabilistic Models to Infer Infection Rates in ViralOutbreak www.stats.ox.ac.uk/__data/assets/file/0013/3352/infection_rates.pdf
[14] Operations Research: Theory and Applications 5th Edition by Sharma J K
[15] Probability and Statistics with. Reliability ... by K.S. Trivedi
[16] B. Aditya Prakash, Hanghang Tong, Nicholas Valler, Michalis Faloutsos, and Christos Faloutsos "Virus Propagation on Time-Varying Networks: Theory and Immunization Algorithms"
[17] An Integrated System for Regional Environmental Monitoring and Management Based on Internet of Things by Shifeng Fang, Li Da Xu, Senior Member, IEEE, Yunqiang Zhu, Jiaerheng Ahati, Huan Pei, Jianwu Yan, and Zhihui Liu IEEE transactions on industrial informatics, vol. 10, no. 2, May 2014
[18] Lucas D. P. Mendes, Joel J. P. C. Rodrigues, Senior Member, IEEE,Jaime Lloret, Senior Member, IEEE, and Sandra Sendra"Cross-Layer Dynamic Admission Control for Cloud-Based Multimedia Sensor Networks"IEEE systems journal, vol. 8, no. 1, March 2014.
[19] Fang Zhaho, Leonidas Guibas, "Wireless Sensor Networks: An information Processing Approach", Elsevier ISBN: 978-81-8147-642-5

Licensed Under Creative Commons Attribution CC BY

[20] Kazim Sohraby, Daniel Minoli, Taieb Znati, "Wireless Sensor Networks: Technology,
[21] Protocols and Applications", Wiley ISBN: 978-81-265-2730-4 (Students Edition)
[22] Remote Sensing and Geographical Information system byM.AnjiReddy
http://en.wikipedia.org/wiki/Dijkstra%27s_algorithm

Author Profile



Sulochana Gore has received her B.E. degree in Computer Science and Engineering from Terna College of Engineering, Osmanabad, Aurangabad University, Aurangabad, Maharashtra State, India, in 2006. She is currently pursuing her M.E. degree in Computer Engineering from Rajarshi Shahu School of Engineering and Research, Narhe, Pune, Pune University, Maharashtra State, India. During the period 2007 to 2013she has worked as a lecturer in Aurangabad and Pune University in the department of Computer Engineering.



Dr. Sulochana B. Sonkamble has received her B.E., M.E. and Ph.D. degrees in Computer Science & Engineering from Shri Guru Govind Saheb Institute of Engineering & Technology, Nanded, Swami Ramanand Teerth Marathwada University, Nanded, Maharashtra State, India. She is currently working as Head of the Computer Engineering Department Rajarshi Shahu School of Engineering and Research, Narhe, Pune, Pune University, Maharashtra State, India.

Licensed Under Creative Commons Attribution CC BY