Perspectives in Communication, Embedded-Systems and Signal-Processing (PiCES) – An International Journal
ISSN: 2566-932X, Vol. 1, Issue 5, August 2017
Proceedings of National Conference on Emerging Trends in VLSI, Embedded and Networking (NC-EVEN 17), May 2017

A Fault Tolerant Approach to Extend Network Life Time of Wireless Sensor Network

Raghavendra Rao N S
4th Sem, M. Tech, RNSIT, Bangalore,
raghavendraraons22@gmail.com

Andhe Pallavi
Professor & HOD, EIE Dept, RNSIT, Bangalore

Abstract: In a wireless sensor network the delivery of the data packet from source to destination may be failed for various reasons and major due to failure-prone environment of networks. This may happen due to the topology changes, node failure due to battery, exhaust or breakdown of the communication module in the wireless node and results in the link failure. This paper addressed the major problem of link failure due to the failure of the nodes in the WSN and with the aim of providing robust solutions to satisfy the QoS-based stern end-to-end requirements of communication networks. In this paper, we propose the new solution by modifying the existing extended fully distributed cluster-based routing algorithm (EFDCB). In this proposed algorithm the faulty nodes or nodes that are more prone to failure in the every cluster of the network get identified by exchanging data and mutually testing among neighbour nodes. When we established the path between source and destination these faulty nodes get excluded in the path selection process and more stable, less prone to failure path will be formed. The performance of this new modified fault-tolerant fully distributed cluster-based routing algorithm is evaluated by simulating it in NS2 environment. Simulation results show that it performs better than the existing algorithm and provide novel solution for fault detection and fault management along the QoS paths and achieves a high degree of fault tolerance.

Keywords: Wireless sensor network; fault detection; fault management; fault tolerance

I. INTRODUCTION

A wireless sensor network consists of a group of homogeneous or heterogeneous nodes that form an arbitrary network interconnected via by means of several wireless communication media without any fixed infrastructure and is used for monitoring distributed remote environments. In such networks, sensor nodes that coordinate with each other to collect the data from the environment such as temperature, humidity, sound, etc. and then send it to the main station. The data collected by the sensors are transmitted over a wireless channel to a running application or base station that makes decisions based on these data after performing the analysis. Wireless sensor nodes are usually mass produced and are often deployed in remote and unfriendly environments making them more prone to failures than any other systems. Again, due to arbitrary deployment some of the nodes may get mobile which leads to topology changes and problem of link breakage get occurred between the source and destination. Traditionally, these problems get solved by reestablishing [5] the connections from source to destination and reroute the packets. This approach creates an additional burden on network management and a lot of battery power of sensor node gets wasted and the network lifetime gets reduced. To solve this problem, the author [1] [2] proposed local repair of the connections by intermediate nodes, then the associated connections will likely only suffer minor disturbance. This approach works fine and has considerable benefits over a global alternative only when if the rate of failure of the nodes is lower and numbers of nodes within the cluster will be more. If the rate of node failure is high, then problem of link failure will be worse and most of the node energy gets wasted in the reestablishment of the path which lowers the network lifetime, increases the drops, and reduces the packet delivery ratio and throughput.

The majority of applications proposed for wireless sensor network have specific quality of service (QoS) requirements and it requires a continuous fault free operation. In order to meet these requirements reliably, WSNs require effective fault management with a high degree of fault-tolerance (FT) and fast recovery time when links get fail on an intermittent or permanent basis or try to establish the link which retain for the longer duration to achieve the QoS requirements of wireless sensor networks.

This paper presents a novel approach for the fault detection and fault management in the wireless sensor network. In this proposed clustered approach, after forming the clusters in the network, we find out faulty or probably faulty sensor node by applying the neighbor coordination within the cluster and then propagating the information about faulty nodes to all the cluster head. When network tries to establish the path between a source node and destination node these faulty nodes get excluded in the path selection process. As faulty nodes already excluded from the path selection process the path will be maintained for the longer duration, also when the link gets broken due to mobility of nodes, the link gets reestablished by applying the local repair and while reestablishment the path faulty node again get excluded.

The proposed method gives the high degree of fault tolerance in real sense.

II. Related Work

In this section, we take a brief review of the works that is related to the domain of fault detection and fault tolerance in WSNs. Because of faulty sensors in WSN, network quality of service will get degrade and create the huge burden on the limited energy of the sensor node.

A weighted median fault detection scheme (WMFDS) is proposed [10] in which the author uses the spatial correlation of sensor measurements for the detection of faulty nodes in the network and it was evaluated for both binary decisions and real number measurements by weighting the neighbor's measurement.

For eliminating unpredictable local decisions while performing Distributed decision fusion author [11] proposed the Collaborative sensor fault detection (CSFD) scheme. Depending on the pre-designed fusion rule and assuming equal local decision rules in fault-free environments author has established an upper bound on the fusion error probability. A criterion has been proposed to find out the faulty nodes according to this error boundary and all local decisions are removed from the computation of the probability ratios that are adopted to make the final decision.

The authors [13] proposed a localized fault detection method to find out the faulty sensors in the network. In this distributed fault detection (DFD) method the fault detection accuracy will decrease rapidly as the number of neighbor nodes to be diagnosed is all small and the node's failure ratio is high. High fault detection accuracy can be achieved only when it is applied to the sensor network with many neighbors of nodes to be diagnosed.

The author [12] proposed an improved DFD scheme is by defining new detection criterion to remedy the shortcomings of [13] original DFD. Improved DFD scheme increases fault detection accuracy greatly. The improved DFD algorithm finds out the faulty nodes even if number of neighboring nodes is less in number.

To transmit the data packets from source to destination many QoS routing algorithms have been proposed. Local Proportional Sticky Routing (PSR) was the first localized QoS routing scheme [7], [18]. PSR is used as an alternative to global QoS routing. PSR operates in two steps: proportional flow routing and computing flow proportions. Proportional flow routing determines the path of traffic during a cycle. When a cycle is complete, a new flow proportion is found for each path based on blocking probabilities.

Credit-Based Routing [4] uses a credit scheme in which rewards will be given to path for flow acceptance and penalizes for the rejection. Path selection is based on path credits where higher credit paths are preferred. CBR also monitors flow blocking probabilities for each path to use in future paths.

Quality-Based Routing [8], [18] determines paths based on QoS metric values. QBR monitors a path and translates flow values into average path qualities. QBR rewards successful flow and punishes flow error like CBR. The difference is that CBR assigns credits based on blocking probabilities while QBR uses average path quality. Delay-Based QoS Routing [3][18] uses the average delay on a path to make its routing decisions. DBR uses average path delay to measure the path's quality and upon flow arrival, the path with the least average delay is used to route the incoming traffic.

Stable and Delay Constraints Routing [6] works in two steps: routing discovery and maintenance. Link stability and delay constraints are also considered in the two steps. In the discovery step, it sends a QoS request to the destination first, and selects the most stable path. If there is no stable path, it will broadcast a route request (RREQ). When the source receives a route report (RREP), it will calculate end-to-end delays and determine the best path. If it receives a route error message (RRER), it will delete that route from its cache. It will then recalculate the best route for traffic.

QoS routing [1] [2] can be centralized, distributed, or hierarchical. Centralized routing requires that nodes maintain global knowledge at the source. The global state has to be updated frequently to cope with network dynamics. Distributed routing algorithms can be more scalable since path computation is divided among the nodes. Many distributed schemes make routing decisions hop-by-hop, but rely on global state for QoS routing. Hierarchical routing shares best features of both centralized and distributed schemes. Every node maintains fractional knowledge of network state. Groups of nodes are aggregated for scalability. Source routing occurs at each hierarchical level to find feasible paths, with some inaccuracy

Nargunam and Sebastian's fully distributed cluster-based (FDCB) [5] algorithm addresses QoS routing. With FDCB, scalability issues in centralized routing are circumvented. The FDCB method is similar to hierarchical routing in that each cluster node only maintains QoS information for other cluster members, a fraction of the network. Thus, an increase in nodes should not significantly increase memory or runtime. Further, since the global network state is shared and maintained by all, the communication overhead is greatly reduced. In FDCB, if the path from source to destination is not in the same cluster, the source sends a route request packet to the gateway node, which forwards it to adjacent cluster(s). As long as the intermediate gateway nodes and links can support the requested QoS constraints, this process is repeated until the destination is found. The discovered path is then sent back to the source and the resource reservation made. The distributed nature of FDCB allows it to avoid unmanageable shared global state. FDCB's distributed routing adds initial latency for the route discovery.

Route requests may not flood the network due to its clustered architecture, but precautions are needed to ensure route queries propagate efficiently from source to destination. Each cluster in the FDCB algorithm has the potential to obtain gateway nodes, which maintain communication with adjacent clusters. With FDCB there is no need for the node aggregation used in hierarchical routing since clusters need not be represented by an aggregate data structure. Although FDCB addresses many of the difficulties with traditional QoS routing schemes, it employs a distributed routing method, which has significant drawbacks. The paper does not discuss how failures are handled. Support for cluster joins and leaves are provided; however, the problem of mitigating the impact on QoS in the event of an unpredicted node leave/failure is untreated. This event is presumably handled by the common practice of rerouting QoS traffic from the source. Nargunam and Sebastian [5] illustrate the problems with conventional clustering where each cluster has only one node, the "cluster-head," responsible for organizing and maintaining the cluster. Traditional cluster construction requires a cluster head election each time one fails or leaves. If the cluster-head fails or leaves, all of its information and responsibilities become orphaned until a replacement is elected. To avoid this problem, the author [1][2] proposed EFDCB a distributed architecture in which each cluster member maintains a QoS parameter table for each of its cluster members and a table containing all cluster gateway nodes and connection failures are handled by applying local repair at the site closest to the link breakdown. This local recovery method only suffers minor disruption to the path. The EFDCB addresses solution to the many shortcomings of original FDCB algorithm but it will find the alternate path by applying the local repair only when the link gets broken. For the reestablishment of the path it takes some time and during that period considerable amount of packets are dropped and node energy gets wasted for reestablishment of connections. Again rerouting the traffic from the source or local repair gives up serious effect on network lifetime when node failure ratio is high or multiple paths using the failed node by increasing the communication overhead for frequent reestablishment of the connection. It does not discuss any mechanism to find the faulty nodes in the network prior to path selection process. Again, there is no effective mechanism to check the node status, whether it is good or faulty.

In this paper, we eliminate the faulty nodes or nodes which are more prone to failure within the cluster and then we find out the path for routing the data packets. The proposed method used the best features of [1] [12] and provide the robust solution to handle the link breakage in WSN.

III. Problem Definition

A. Introduction

In this paper we mainly concentrate on the fault tolerance problem during the link failure in WSN and designed a robust solution to maintain QoS requirements. In the preceding section we discussed modified [12] technique that has offered, efficient mechanism to find out the faulty node in the network by using the neighbor coordination and exchanging the QoS parameters. The problem of link failure in the WSN is handled by applying the local repair [1] at the site closest to the link breakdown. The method presented here is the use of the best features of these two different methods to achieve a high degree of fault tolerance. This section presents the motivation, definitions, and assumptions for the proposed algorithm, which is a fault-tolerant modification to EFDCB routing [5].

B. Aims and Premise

Nargunam and Sebastian [5] propose a fully distributed algorithm, FDCB, in which clustering provides scalability by lowering the amount of information maintained at each node. FDCB addresses the scalability issue effectively, but fails to maintain QoS connections when nodes move, leave the network, or fail. As the authors provide no details, it is assumed that when a QoS path suffers a link breakage, the source reroutes the traffic via a completely new path by negotiating the QoS Parameters. FDCB also uses a distributed reactive routing technique like AODV, DSR which causes undesired packet transmission latency for QoS routing applications. FDCB does not provide a feasible routing scheme or local fault tolerance, but serves as base to that end. EFDCB [1][2] extends FDCB to provide local repair of the link failure by considering the fraction of the total network link when finding a new feasible path through local recovery in the cluster. Hence, the load of negotiating newly calculated QoS paths, as is done in rerouting by FDCB, is somewhat reduced. The EFDCB addresses solution to the many shortcomings of original FDCB algorithm and it will find the alternate path by applying the local repair when the link gets broken. Again for reestablishment of the path it takes some time and during that period considerable amount of packets is dropped and node energy gets wasted for reestablishment of connections. It does not discuss any mechanism to find the faulty nodes in the network prior to the path selection process so as it is excluded in path selection process. For this reason, the new method is expected to have a considerable prior identification of faulty nodes within the cluster and avoid such nodes from the path selection process to get run time advantage resulting in improved long lasting QoS route. Faster QoS recovery time equates to lower QoS disruption time, improved packet delivery ratio, fewer dropped packets, improved throughput and average network energy resulting in the extended network lifetime.

C. Approach

In [24] proposed the method to find out the faulty node with greater accuracy even if node failure ratio is high and number of neighbor nodes is less in the distributed wireless environment. To find out the faulty nodes within the cluster we have to use similar a approach as proposed by Peng Jiang [12]. The information about faulty node propagates within the network by cluster head. Similar to EFDCB in this approach we use Table-Driven or Proactive routing protocols such as DSDV [19] so that the cluster-head has the complete "cluster-state" knowledge. The cluster-head has knowledge of connectivity awareness for all cluster nodes. This awareness includes knowledge of all QoS connections currently supported by each cluster member, each member's resource availability, faulty sensor nodes and the cluster topology. In this proposed system, when cluster node A leaves the cluster, due to mobility or failure, and the QoS paths supported by A are broken, the cluster-head has all required information to begin a renegotiating to reestablish the connection with minimum delay by excluding the faulty nodes. The cluster head collects this knowledge via two processes: communication with the other clusters via clustered FSR and local clustered information exchange. These processes guarantee, with high probability and low overhead, that knowledge of the systems' state is maintained both to repair existing paths and to initiate new ones by excluding the faulty. As we exclude the faulty nodes or nodes which are prone to failure the selected path will be stable for the longer duration, which reduces the communication overhead, results increase in the network lifetime.

D. Proposed System

The proposed approach is targeted toward routing QoS packets in challenging WSN environments where nodes often get fail and due to which links can often break without warning. In these environments, a routing algorithm needs a contingency plan for link breakages. In this system we modify the existing EFDCB algorithm such that before transmitting the data it first finds out the faulty nodes in the every cluster and propagate this information to all the cluster head. Similarly for repairing the broken path at the failed link we shift the traffic to a neighboring fault free node and then routing around the breaking point. This method avoids the costly process of rerouting also avoiding the faulty node.

Assumptions:

- Every node has an exclusive identifier.
- Two nodes can be members of the same cluster if their Euclidean distance is <= 250m.
- Nodes show their existence via a periodic beacon Message.
- Path routing table entries for all network destinations is maintained by all gateway nodes in the network.

IV. Simulation Tool, Implementation, Result

A. Simulation Tool

Network Simulator (NS2) is a discrete event driven simulator tool which is developed at UC Berkeley. Basically now it is part of the VINT project [9]. The main goal of NS2 is to support networking research and education. It is mainly suitable for designing new protocols, comparing different protocols and for traffic evaluations in the wireless scenario. NS2 implement network protocols such as UPD and TCP, traffic source behavior such as Telnet, FTP, Web, VBR and CBR, router queue supervision mechanism such as Drop Tail, RED and CBQ, routing algorithms such as Dijkstra, and many more. NS2 also implements multicasting and some of the MAC layer protocols for LAN simulations. The functions of a Network Simulator are to create the event scheduler, to create a network, for computing routes, to create connections, to create traffic and processed in the order of their scheduled Occurrences. NS2 is also useful for inserting errors and tracing can be done with it. Tracing packets on all links by the function trace-all and tracing packets on all links in nam + format using the function nam trace-all [9].

A large amount of institutes and people in development and research use, maintain and develop NS2. This increases the confidence in it. It is distributed freely and open source. Versions are available for FreeBSD, Linux, Solaris, Windows and Mac OS X [9].

B. Simulation Parameters

In the simulation, 46 mobile nodes move in a 2000 meter x 2000 meter region for 20 seconds of simulation time.

No. of Nodes	46
Area Size	2000 X 2000
Mac	IEEE 802.11
Transmission Range	40 m
Traffic Source	CBR
Packet Size	500
Initial Energy	100 J
Transmission Power	1.0
Receiving Power	0.5

Table 1. Simulation Parameter

Transmission range of all nodes is 40 meters. The simulation parameters and settings are shown in Table 1.

C. Implementation of Proposed Approach

To achieve efficient fault tolerance, our proposed approach is broadly divided into four different stages:

1. Cluster formation
2. Selection of Cluster Head
3. Searching for faulty nodes or nodes prone to failure within the cluster
4. Selection of Fault free path for data transmission

a) Cluster Formation

In this stage we form the clusters from the randomly deployed sensor nodes using neighbor coordination and are responsible for establishing and maintaining neighbor relationships. Neighbors can meet each other simply by transmitting a special packet (HELLO packet) over the broadcast medium. In the wireless network, HELLO packets are periodically broadcasted and nodes within the transmission range of the sending node will hear these special packets and record them as neighbors.

Perspectives in Communication, Embedded-Systems and Signal-Processing (PiCES) – An International Journal
ISSN: 2566-932X, Vol. 1, Issue 5, August 2017
Proceedings of National Conference on Emerging Trends in VLSI, Embedded and Networking (NC-EVEN 17), May 2017

Each node associates a TIMEOUT value in the node's database for each neighbor. When it does not hear a HELLO packet from a particular neighbor within the TIMEOUT period, it will remove that neighbor from the neighbor list.

b) Selection of Cluster Head

In this stage, each node within the cluster broadcast its energy level and its position to the every other node of the cluster. After this broadcast, all the nodes within the cluster get the energy level and position of each and every node of the cluster. Now nodes will arrange the received energy level of other nodes in the ascending order. The average energy of the cluster is calculated and nodes whose energy level is less than the average energy level of the network will be discarded from the cluster head nomination criteria. The node with highest energy level within the cluster is declared himself as a Cluster Head (CH) and this information get shared with all the members.

c) Identification of Faulty Nodes

I In this stage, the faulty or likely to be faulty nodes are identified by exchanging the information such as energy, various sensors reading with the neighboring nodes within the cluster. The parameters which are used for selection of the faulty nodes depends on the application and specific condition of the network, accordingly we will set the threshold limit. After identification, the information about the faulty nodes gets shared with all the cluster heads.

d) Selection of Fault Free Path

When network tries to establish the path for transmitting the data from source to the destination these faulty nodes get eliminated in the path selection process. As the faulty nodes get eliminated path is more stable for the longer duration. If the link gets broken due to the mobility or some reason, then by applying the local recovery link get reestablish and again faulty nodes get excluded from the reestablishment process. Every time the faulty nodes get excluded from the path selection process, the path will be more stable for the longer duration. By applying pre diagnosis of faulty nodes and eliminating it in path selection, communication overhead gets greatly reduced as the network has to repair the path less number of times. Reduced communication overhead leads to saving of precious limited battery power of the nodes and result into an extension of network lifetime.

D. Simulation Result

Figure 1 shows the packet delivery ratio for the modified EFDCB, EFDCB and FDCB at a different simulation time. As the path selected by omitting the faulty nodes or nodes that are prone to failure simulation shows the improvement in the packet delivery ratio than that of EFDCB and FDCB.

Figure 2 shows the packet drops for the modified EFDCB, EFDCB and FDCB at a different simulation time. As path is selected by omitting the faulty nodes or nodes that are prone to failure simulation shows the no packet drops than that for EFDCB and FDCB.

Figure 3 shows the average energy consumption of the network for the modified EFDCB, EFDCB and FDCB at a different simulation time. As the path is selected by omitting the faulty nodes or nodes that are prone to failure, simulation shows the considerable improvement in average energy consumption of the network. It is observed that the average energy consumption of modified EFDCB is better than that of EFDCB and FDCB. It means network lifetime will be extended as more energy gets conserved.

Figure 4 shows the throughput for the modified EFDCB, EFDCB and FDCB at a different simulation time. As the path selected by omitting the faulty nodes or nodes that are prone to failure simulation shows the improvement. It is observed that the throughput get increased than that of EFDCB.

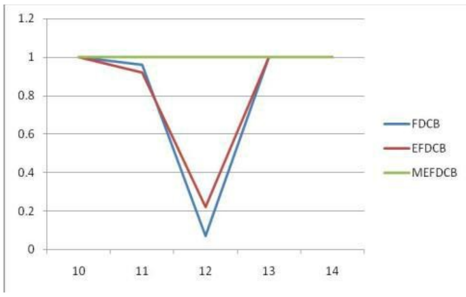

Fig 1. Packet Delivery Ratio

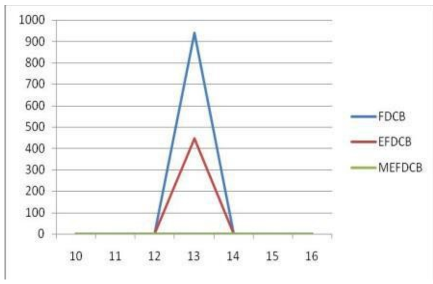

Fig 2. Packet Drops

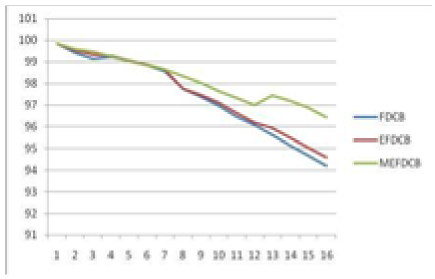

Fig 3. Average Energy

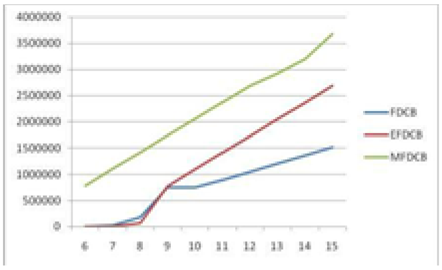

Fig 4. Throughput

V. CONCLUSION

This paper proposes a modified fault tolerant distributed routing algorithm for the link failure issues in the wireless sensor networks. Link failure occurs due to various reasons like the mobility of sensor nodes, battery depletion or communication module failure. The work presented in the paper shows that the traditional method of rerouting traffic from the source or local repair gives up serious effect on network lifetime when node failure ratio is high or multiple path using the failed node. It also shows that, if we first find out the faulty nodes or nodes that are more prone to failure in the network and excluding them in the path selection process then there is positive effect on node energy by reducing the communication overhead. This method conserves the node energy and extends the network lifetime. It reduces the packet drops by improving the packet delivery ratio and improves the throughput. The proposed method is more useful for the network where problem of link failure is worst due to mobility or node failure.

REFERENCES

[1] Larry C. Llewellyn, Kenneth M. Hopkinson, Member, IEEE, and Scott R. Graham, Member, IEEE," Distributed Fault-Tolerant Quality of Wireless Networks" IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 10, NO. 2, FEBRUARY 2011

[2] R. V. Kshirsagar, A.B. Jirapure," An Approach for fault detection and fault management in WSN to extend network lifetime's, European Scientific Journal ,vol.11, No. 12, pp 326-336, April 2015

[3] A.S. Alzahrani and M.E. Woodward, "End-to-End Delay in Localized QoS Routing," Proc. IEEE Int'l Conf. Comm. Systems (ICCS), pp. 1700-1706, 2008

[4] S.H. Alabbad and M.E. Woodward, "Localised Credit Based QoS Routing," IEE Proc.—Comm., vol. 153, no. 6, pp. 787-796, Dec. 2006

[5] A.S. Nargunam and M.P. Sebastian, "Fully Distributed Cluster Based Routing Architecture for Mobile Ad Hoc Networks," Proc. IEEE Int'l Conf. Wireless and Mobile Computing, Networking, and Comm., pp. 383-389, 2005.

[6] P. Yang and B. Huang, "QoS Routing Protocol Based on Link Stability with Dynamic Delay Prediction in MANET," Proc. Pacific- Asia Workshop Computational Intelligence and Industrial Applications (PACIIA), pp. 515- 518, 2008.

[7] S. Nelakuditi, Z.L. Zhang, R.P. Tsang, and D.H.C. Du, "Adaptive Proportional Routing: A Localized QoS Routing Approach," IEEE/ ACM Trans. Networking, vol. 10, no. 6, pp. 790-804, Dec. 2002

[8] A.H. Mohammad and M.E. Woodward, "Localized Quality Based QoS Routing," Proc. Performance Evaluation of Computer and Telecomm. Systems (SPECTS), pp. 209-216, 2008

[9] Mamatha B L, Ch. Sudershan Raju, T.Kantharaju, "Co-operative opportunistic routing in VANETS using PSR (Proactive Source Routing Protocol)" IJREAT, Volume 2, Issue 2, Apr-May, 2014

[10] Gao, J.L.; Xu, Y.J.; Li, X.W. Weighted-median based distributed fault detection for wireless sensor networks. J. Softw. 2007, 18, 1208-1217.

[11] Wang, T.Y.; Chang, L.Y.; Dun, D.R.; Wu, J.Y. Distributed fault-tolerant detection via sensor fault detection in sensor networks. In Proceedings of the IEEE 10th International Conference on Information Fusion, Quebec, Canada, 2007; pp. 1-6.

[12] Peng Jiang, A New Method for Node Fault Detection in Wireless Sensor Networks, Sensors 2009, 9, 1282-1294.

[13] Chen, J.R.; Kher, S.; Somani, A. Distributed fault detection of wireless sensor networks. In Proceedings of the International Conference on Mobile Computing and Networkings, Los Angeles, CA, USA; ACM: New York, USA, 2006; pp. 65-72.

[14] S. Chen and K. Nahrstedt, "Distributed Quality-of-Service Routing in Ad-Hoc Networks," IEEE Selected Areas in Comm., vol. 17, no. 8, pp. 1488-1505, Aug. 1999

[15] S. Chakrabarti and A. Mishra, "QoS Issues in Ad Hoc Wireless Networks," IEEE Comm. Magazine, vol. 39, no. 2, pp. 142-148, Feb.2001

[16] G.V. Arvindh and R. Prem Kumar," Fully Distributed Cluster based Fault Tolerant Service, International Conference on Computing and Control Engineering (ICCCE 2012), 12 & 13 April, 2012

[17] Gayathri V, Sabu Emmanuel, Shrikanthan," Cluster based approach to fault detection and recovery in WSN", IEEE ISWSC, 2007, pp 35-39

[18] Udaya Kumar. N, Suresh.S," Localized QOS Routing in Mobile Adhoc Networks", IJCSM Research Vol 2 Issue 2 Feb 2013

[19] Vishal Garge, Chandan Kapoor," A Survey of QoS parameters through reactive routing in MANETs", IJCEM, ISSN 2230-7893, Vol 13,July 2011 pp