AN APPROACH FOR FAULT DETECTION AND FAULT MANAGEMENT IN THE WIRELESS SENSOR NETWORK TO EXTEND NETWORK LIFETIME

Dr. R. V. Kshirsagar
A. B. Jirapure
Priyadarshini College of Engineering, Nagpur, India

Abstract

A mobile wireless ad hoc sensor network (MANET) consists of a group of homogeneous or heterogeneous mobile communicating hosts that form an arbitrary network interconnected via by means of several wireless communication media without any fixed infrastructure. In such network the delivery of the data packet from source to destination may fail for various reasons and major due to failure-prone environment of networks. This may happens due to the topology changes, node failure due to battery exhaust, failure of the communication module in the wireless node and results in the link failure. This paper addressed the major problem of link failure in the WSN and with the aim of providing robust solution so as to satisfy the stern end-to-end requirements of QoS-based communication networks. In this paper we modifies existing fully distributed cluster-based routing algorithm by addressing local recovery for the link failure. Performance of this new fault-tolerant fully distributed cluster-based routing algorithm is evaluated by simulating it in NS2 environment and we show that it performs better than the existing algorithm and provide better solution for fault detection and fault management along the QoS paths.

Keywords: Wireless sensor network, fault management, fault tolerance

Introduction

Wireless sensor network (WSNs) consists of spatially distributed autonomous sensor nodes that collaborate with each other to collect the environmental data. Typical examples include temperature, light, sound, and humidity. These sensor readings are transmitted over a wireless channel to a running application that makes decisions based on these sensor readings. WSN sensor nodes are typically mass produced and are often deployed in unattended and hostile environments making them more susceptible to failures than other systems. Additionally, due to random deployment some of the nodes may get mobile which leads to topology changes and problem of link failure occur, this may reestablish the connections from source to destination and reroute the packets from source to destination. This create additional burden on network management and lot of battery power of sensor node get wasted due to which the life time of a WSN reduces.

Many applications have been proposed for wireless sensor networks, and many of these applications have specific quality of service (QoS) requirements. Most of the applications in the WSN are mission-critical, requiring continuous operation and in order to meet application requirements reliably, WSNs require effective fault detection, fault-tolerance (FT) and fast recovery when links fail on an intermittent or permanent basis to achieve the QoS requirements of wireless sensor networks.

Most protocols for multi hop WSN routing maintain best effort routes. Excessive node mobility can lead to topology changes before network updates can propagate. Chakrabarti and Mishra call this combinatorial stability. Only combinatorially stable networks are considered in this paper.

In this paper we address stability and recoverability, two main problems in routing QoS traffic in wireless sensor networks.

The first issue is stability. With most ad hoc wireless networks that support QoS, each node acts as a router. In many distributed reactive routing schemes, if a node does not know the QoS parameters of its neighbors it broadcasts the route request packet and the neighboring nodes share their QoS parameters using broadcast packets. The broadcast packets used to discover the QoS parameters of nodes' neighbors and negotiate QoS paths this can flood the network. By using a clustered approach we can lower this communication overhead to more scalable levels by limiting inter cluster control communication to gateway nodes.

The second issue is minimizing the QoS impact due to network failures. When traffic is routed through multiple hops and if a supporting node fails due to some reason then, in the worst case, the connection must be reestablish and data packet rerouted from the source. This global fault-recovery method requires that the source renegotiate a new QoS path, which is costly in computation and communication as node battery power is limited. If multiple sources were using the failed node in QoS paths then each source must negotiate a new path. Despite its cost, this failure method appears to be common. By contrast, if a protocol itself allows intermediate nodes to repair connections locally then the associated connections will likely only suffer minor disruption. (Larry C. Llewellyn, Kenneth M. Hopkinson, 2011) shows local repair can make the difference in meeting time-sensitive deadlines. Simulation of prototype, in this paper, show that a local fault-tolerant algorithm has significant benefits over a global alternative. The local method used is similar to Chen and Nahrstedt's repair algorithm where QoS connection failures are handled at the site closest to the link breakpoint. While the two protocols share this similarity, there are major differences in the repair methods used and the fault tolerance achieved. This paper presents the key features, definitions, and assumptions and implementation of the fault tolerant fully distributed cluster-based (FT-FDCB) routing algorithm, which is a fault-tolerant extension to FDCB and EFDCB (Larry C. Llewellyn, Kenneth M. Hopkinson, 2011).

2. Related Work

Many QoS routing algorithms have been proposed. Local Proportional Sticky Routing (PSR) was the first localized QoS routing scheme (S. Nelakuditi, Z.L. Zhang, R.P. Tsang, and D.H.C. Du, 2002). PSR is simple yet stable and is used as an alternative to global QoS routing. PSR operates in two stages: 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 (CBR) uses a credit scheme that rewards a path for flow acceptance and penalizes 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 (S.H. Alabbad and M.E. Woodward, 2006). Quality-Based Routing (QBR) (A.H. Mohammad and M.E. Woodward,2008) 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 (DBR) (A.S. Alzahrani and M.E. Woodward, 2006) uses the average delay on a path to make its routing decisions. The average path delay is used to measure the path's quality, and, upon flow arrival, the path with the least average delay is used to reroute the incoming traffic.

Stable and Delay Constraints Routing (SDCR) (P. Yang and B. Huang, 2008) works in two major phases: routing discovery and maintenance. Link stability and delay constraints are considered in the two phases. When in discovery, 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 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 advantages of both centralized and distributed schemes. Each node maintains partial network state. Groups of nodes are aggregated for scalability. Source routing occurs at each hierarchical level to find feasible paths, with some inaccuracy

3. Quality of Services Routing Using FDCB

Nargunam and Sebastian’s fully distributed cluster-based (FDCB) (A.S. Nargunam and M.P. Sebastian, 2005) algorithm addresses QoS routing in MANETs. 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 global network state is shared and maintained by all, the communication overhead is greatly reduced. In FDCB, if a flow’s source and destination are 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 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 illustrate the problems with conventional clustering where each cluster has exactly one node, the "cluster-head," responsible for organizing 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, we propose 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. FDCB has no effective way to handle connection failures. The distributed routing design presented is also ill-suited for MANET QoS applications. FDCB provide a path to developing a scalable QoS routing solution, but it could be improved in challenged environments

4. Fault Tolerance in QoS Ad-Hoc Networks

Chen and Nahrsted propose fault tolerance techniques to reduce the impact of QoS disruptions due to link failures or network dynamics. The authors only consider applications which do not require hard guarantees. Further, the authors state that many multimedia applications accept soft QoS and use adaptation techniques to reduce the level of QoS (S. Cen, C. Pu, R. Staehli, C. Cowan, and J. Walpole,1995, N. Tran and K. Nahrstedt, 1998, F. Goktas, F.M. Smith, and R. Bajcsy, 1997) disruption. One technique is to repair the broken path at the failed node by shifting traffic to a neighboring node and then routing around the breaking point. This method avoids the costly process of rerouting the traffic from the source. The second technique uses multilevel path redundancy, which establishes multiple paths for the same connection. First-Level Redundancy sends all data along all paths independently, which is used for "critical" QoS. Second-Level Redundancy sends data along the primary path and only uses secondary paths if the primary path is lost. It is used for connections which can tolerate a degree of QoS failure. Third-Level Redundancy is like second level except the secondary paths are not reserved; only calculated. On failure, an attempt is made to reserve the secondary path. When case 2 occurs, the preceding node broadcasts a repair-request message to its neighbors asking if any of them can take over for the defunct intermediate node. The neighbors that have links to the successive node reply with their resource availabilities to the preceding node. If, based on the replies, the preceding node finds node i has sufficient resources for that role, it adds the link from itself to node i to the routing path and then sends i a path-repair message. When i receive the path-repair message, it reserves the required resources and adds the link from itself to the successive node to the routing path. Once the path has been repaired, a path validation message is sent to insure that the repaired path does not violate its end-to-end constraints. A path-validation message is sent to the destination which sends the message to the source. The source checks to see if the end-to-end requirements have been violated. If they have, the source reroutes the traffic or QoS negotiation with the user application takes place.

5. Problem Definition

5.1 Introduction

This paper focus on the fault tolerance problem in MANETs designed to support QoS requirements. The previous section discussed two techniques that have been offered (S. Chen and K. Nahrstedt, 1999)], a path redundancy technique and a local repair algorithm. The protocol presented here is similar to the local repair algorithm by Chen and Nahrstedt where QoS connection failures are handled at the site closest to the link breakpoint. While the two protocols share this similarity, there are significant differences in the repair methods used and the level of fault tolerance achieved. This section presents the motivation, definitions, and assumptions for the FT- FDCB routing protocol, which is a fault-tolerant modification to FDCB routing (A.S. Nargunam and M.P. Sebastian, 2005).

5.2 Aims and Premise

Nargunam and Sebastian propose a fully distributed algorithm, FDCB, in which clustering provides scalability by lowering the amount of information maintained at each node. FDCB addresses MANET scalability, but fails to effectively maintain QoS connections when nodes move, leave the network, or fail. Since 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. 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 groundwork to that end. FT-FDCB extends FDCB to provide the scalability, efficiency, and fault tolerance critical to maintain QoS connections in a unpredictable wireless environment. The goal is to determine if FT- FDCB provides efficient QoS route recovery by testing it against FDCB. The proposed algorithm only has to consider a fraction of the total number of network links when finding a new feasible path through local recovery in the cluster.

Hence, the burden of negotiating newly calculated QoS paths, as is done in rerouting by FDCB, is significantly reduced. For this reason, the new local method is expected to have a considerable runtime advantage resulting in improved QoS route recovery time. 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 network lifetime.

5.3 Approach

To achieve proficient fault tolerance, modified FDCB uses Table-Driven or Proactive routing protocols such as DSDV so that the cluster-head has the complete “cluster-state” knowledge. The cluster-head has connectivity alertness for all cluster nodes. This alertness includes knowledge of all QoS connections currently supported by each cluster member, each member’s resource availability, and the cluster topology. With this 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 information required to begin a renegotiating to reestablish the connection with minimum delay if possible. 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.

5.4 Proposed System

The proposed algorithm is targeted toward routing QoS packets in challenging WSN environments where links can break often and without warning. In these environments, a routing algorithm needs a contingency plan for link breakages. Technique is to repair the broken path at the failed link by shifting traffic to a neighboring node and then routing around the breaking point. This method avoids the costly process of rerouting the traffic from the source.

Assumptions

. All nodes have a unique identifier.

. Two nodes can be members of the same cluster if their Euclidean distance is <=30m.

. Nodes signal their presence via a periodic beacon Message.

. All gateway nodes in the network have path routing table entries for all network destinations.

6. Simulation Tool, Implementation and Result

6.1 Simulation Tool

Network Simulator (NS2) is a discrete event driven simulator tool which is developed at UC Berkeley. basically it is part of the VINT project (Mamatha B L, Ch. Sudershan Raju, T.Kantharaju, 2014). 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. 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 . 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.

6.2 Simulation Parameters

In the simulation, 46 mobile nodes move in a 2000 meter x 2000 meter region for 20 seconds of simulation time. Transmission range of all nodes is 40 meters. The simulation parameters and settings are shown in Table 1.

No. of Nodes	46
Area Size	2000 X 2000
Mac	IEEE 802.11
Transmission Range	40 m
Simulation Time	20 sec
Traffic Source	CBR
Packet Size	500
Initial Energy	100 J
Transmission Power	1.0
Receiving Power	0.5

Table 1: Simulation Parameter

6.3 Implementation of Prototype

Following design Steps have adopted for the design. We design a wireless sensor adhoc network consists of a group of communicating hosts that form an arbitrary network topology. Mobile nodes send hello packets to their neighbor nodes. The nodes satisfy Euclidean distance condition form the clusters. Nodes with higher energy are selected as Cluster heads. After identifying the cluster head a path is established from a source to destination.

When an intermediate link breaks, few packets will be dropped. To attain efficient fault tolerance, we make cluster-head to have complete “cluster-state” information. The cluster-head has connectivity knowledge for all cluster nodes. This awareness includes knowledge of all QoS connections currently supported by each cluster member, each member’s resource availability, and the cluster topology. Thus, cluster heads allow each node in the network to be aware of the complete network state knowledge.With this scheme, when a link fails in a cluster due to mobility or failure of node, the QoS paths supported by link are broken, the cluster-head has all information required to begin a renegotiating to reestablish the connection with minimal delay. Thus making FT-FDCB different from FDCB since link failures are handled locally instead of rerouting traffic from the source and QoS traffic is routed with lower packet transmission delays.

6.3 Simulation Results

Figure 1 shows the packet delivery ratio for FT-FDCB and FDCB at different simulation time. When the link get fail FT-FDCB shows the improvement in the packet delivery ratio by 15 % than that of FDCB.

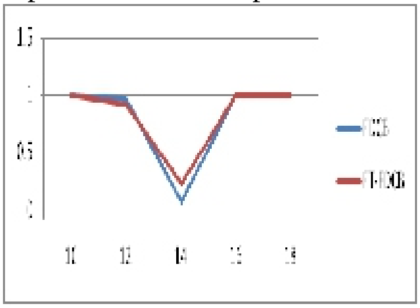

Figure 1: Packet Delivery Ratio

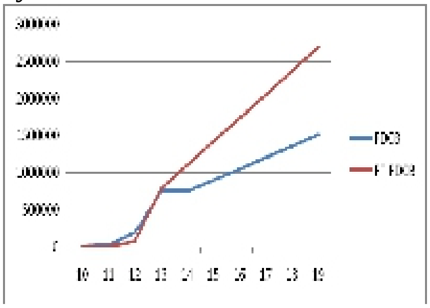

Figure 2: Throughput

Figure 2 shows the throughput for FT-FDCB and FDCB at different simulation time. It is observed that the throughput get increased by nearly 75% than that of FDCB.

Figure 3 shows the average energy consumption of the network for FT-FDCB and FDCB at different simulation time. It is observed that the average energy consumption of FT-FDCB is better than that of FDCB.It means network lifetime will be more.

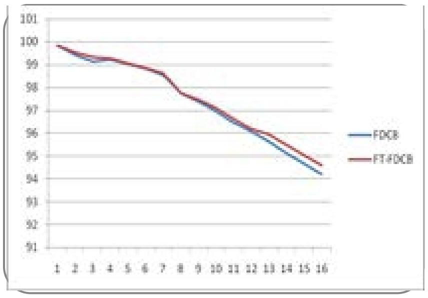

Figure 3: Average Energy

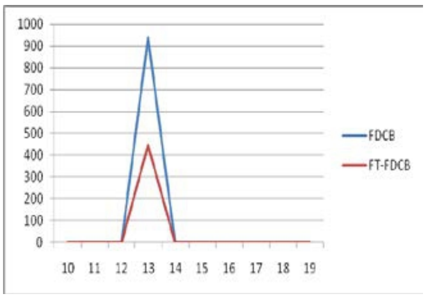

Figure 4: Packet Drops

Figure 4 shows the packet drops when the link get fails for FT-FDCB and FDCB. It is observed that the packet drops get decreased by nearly 50% than that of FDCB.

7. Conclusion

This paper proposes a Fault Tolerant distributed routing algorithm for link failure in the wireless sensor networks. Link failure occurs due to various reason like mobility of sensor node, battery depletion and communication module failure. The work demonstrates that the traditional method of rerouting QoS traffic from the source given a link failure give up serious effect on network life time, through put, packet delivery ratio and drops. It also shows that, if link failure treated locally then there is positive effect on network life time by reducing the drops, improving packet delivery ratio, improving throughput and energy consumption by thenetwork.

Acknowledgement

We are very much thankful to authors Larry C. Llewellyn, Kenneth M. Hopkinson, for their paper titled Distributed Fault-Tolerant Quality of Wireless Networks IEEE TRANSACTIONS

References:

Larry C. Llewellyn, Kenneth M. Hopkinson, (2011), Distributed Fault-Tolerant Quality of Wireless Networks, *IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 10, NO. 2,*

A.S. Alzahrani and M.E. Woodward, (2006), End-to-End Delay in Localized QoS Routinng, *Proc. IEEE Int'l Conf. Comm. Systems (ICCS), pp. 1700-1706*

S.H. Alabbad and M.E. Woodward, (2006), Localized Credit Based QoS Routing, *IEE Proc.—Comm., vol. 153, no. 6, pp. 787-796*

A.S. Nargunam and M.P. Sebastian, (2005), 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*

P. Yang and B. Huang, (2008), 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*

S. Chen and K. Nahrstedt,(1999), Distributed Quality-of-Service Routing in Ad-Hoc Networks, *IEEE Selected Areas in Comm., vol. 17, no. 8, pp. 1488-1505*

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

S. Chakrabarti and A. Mishra, (2001), QoS Issues in Ad Hoc Wireless Networks, IEEE *Comm. Magazine, vol. 39, no. 2, pp. 142-148.*

S. Chen and K. Nahrstedt,(1998), On Finding Multi-Constrained Paths, *Proc. Record 1998, IEEE Int'l Record on Comm. (ICC '98), pp. 874-879*

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

S. Chen,(1999),Routing Support for Providing Guaranteed End-to-End Quality-of-Service, *PhD dissertation, Univ. of Illinois.*

D. Mitra and K.G. Ramakrishnan,(1999), A Case Study of Multiservice, Multipriority Traffic Engineering Design for Data Networks, *Proc. IEEE Int'l Global Telecomm. Conf. (GLOBECOM '99), pp. 1077- 1083.*

D. Applegate and M. Thorup, (2003), Load Optimal MPLS Routing with N þM Labels, *Proc. IEEE INFOCOM*.

S. Cen, C. Pu, R. Staehli, C. Cowan, and J. Walpole,(1995) A Distributed Real-Time MPEG Video Audio Player, *Fifth Int'l Workshop Network and Operating System Support of Digital Audio & Video (NOSSDAV '95), pp. 151-162.*

N. Tran and K. Nahrstedt, (1998), Active Arbitration by Program Delegation in Video on Demand, *Proc. IEEE Int'l Conf. Multimedia Computing and Systems, pp. 96-105*.

F. Goktas, F.M. Smith, and R. Bajcsy,(1997), Telerobotics over Communication Networks, *Proc. 36th IEEE Conf. Decision Control, pp. 2399-2404.*

X. Yuan(2002), Heuristic Algorithms for Multiconstrained Quality-of-Service Routing, *IEEE/ACM Trans. Networking, vol. 10, no. 2, pp. 244-256.*

Xuanwen Luo, Student Member, IEEE, Ming Dong, Member, IEEE, and Yinlun Huang, (2006), On Distributed Fault-Tolerant Detection in WSN, *IEEE TRANSACTIONS ON COMPUTERS, VOL. 55, NO. 1.*

G.V. Arvindh and R. Prem Kumar, (2012), Fully Distributed Cluster based Fault Tolerant Service, *International Conference on Computing and Control Engineering (ICCCE 2012),*

Udaya Kumar. N, Suresh.S, (2013), Localized QOS Routing in Mobile Adhoc Networks, *International Journal of Computer Science and Management ResearchVol 2 Issue 2* ON MOBILE COMPUTING, VOL. 10, NO. 2, 2011

Mamatha B L, Ch. Sudershan Raju, T.Kantharaju,(2014),Co-operative opportunistic routing in VANETS using PSR (Proactive Routing Protocol), *IJREAT, Volume 2, Issue 2,*