The Enhanced Fault-Tolerance Mechanism of AODV Routing Protocol for Wireless Sensor Network

Zamree Che-Aron†, Wajdi Al-Khateeb††, and Farhat Anwar†††

Department of Electrical and Computer Engineering
International Islamic University Malaysia
Kuala Lumpur, Malaysia 50728

Summary

As more and more real Wireless Sensor Network's (WSN) applications are tested and deployed over the last decade, the research community of WSN realizes that several issues need to be revisited from practical angles, such as reliability and availability. Basically, wireless sensor networks suffer from resource limitations, high failure rates and faults caused by the defective nature of wireless communication and the wireless sensor characteristics. This can lead to situations, where nodes are often interrupted during data transmission and blind spots occur in the network by isolating some of the devices. In this paper, we address the reliability issue by designing and developing an enhanced fault-tolerance mechanism of Ad hoc On-Demand Distance Vector (AODV) routing protocol for WSN called the ENhanced FAult-Tolerant AODV (ENFAT-AODV) routing protocol. The proposed ENFAT-AODV routing protocol improves the reliability and robustness of the network by creating a backup path for every node on a main path of data delivery. When the node gets failure to transmit a data packet through the main path, it immediately utilizes its backup route to become a new main path for the transmission of next coming data packets without any interruption. This protocol reduces the number of dropped data packets and maintains the continuity of data packet transmission in presence of network faults. The simulation results prove that the proposed ENFAT-AODV routing protocol enhances the original AODV in term of the reliability, availability and fault-tolerant ability of the network.

Key words:
Wireless sensor network, Fault-tolerance, AODV, Backup path, Routing protocol.

1. Introduction

The advancements in wireless communication technologies enabled large scale wireless sensor networks (WSNs) deployment. A wireless sensor network is a wireless network consisting of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations [1]. The development of wireless sensor networks was originally motivated by military applications such as battlefield surveillance. However, wireless sensor networks are now used in many civilian application areas, including environment and habitat monitoring, healthcare applications, home automation, and traffic control.

Routing in sensor networks is very challenging due to several characteristics that distinguish them from contemporary communication and wireless ad hoc networks. First of all, in contrary to typical communication networks, almost all applications of sensor networks require the flow of sensed data from multiple regions (sources) to a particular sink. Second, sensor nodes are tightly constrained in terms of transmission power, on-board energy, processing capacity and storage and thus require careful resource management [2].

In addition, nodes in WSNs are prone to failure due to physical damage, communication link errors, environmental interference, software bugs, malicious attack, and so on [3]. Moreover, two components of a sensor node, sensing unit and wireless transceiver, usually directly interact with the environment which is subject to variety of physical, chemical, and biological factors. It results in low reliability of performance of sensor nodes. Even if condition of the hardware is good, the communication between sensor nodes are affected by many factors, such as fading, signal strength, obstacles, weather conditions, interference and so on.

Fault tolerance is the ability of a system to deliver a desired level of functionality in the presence of faults [4]. Since the sensor nodes are prone to failure, fault tolerance should be seriously considered in many sensor network applications. Actually, extensive works [5-7] have been done on the issue of fault tolerance and it is one of the most important topics in WSNs.

Currently, there exist several AODV based routing protocol proposals and/or implementations which are suitable or have been specifically designed for the environments of WSN such as AODVjr [8], TinyAODV [9], AODVbis [10], LoWPAN-AODV [11], LOAD [12], NST-AODV [13] and EAODV [14]. To the best knowledge of the authors, we argue that none of previously proposed AODV based routing protocols significantly addresses a fault tolerant issue. All these encouraging statements make the reliable decision to select

Manuscript received June 5, 2010
Manuscript revised June 20, 2010

the AODV routing protocol to be modified for enhancement of fault tolerance in WSN.

In this paper, we propose the ENhanced FAult-Tolerant AODV (ENFAT-AODV) routing protocol which handles the issue of fault tolerance and robustness in wireless sensor network by enhancing the fault tolerance mechanism of AODV (Ad hoc On Demand Distance Vector) routing protocol [15]. We design the fault tolerance mechanism by creating the backup route for all nodes on the main path of data delivery. When the node fails to deliver the data packet through the main route, then it immediately utilizes its backup route to transmit the next coming data packets instead of the previously broken route without any interruption of data transmission to reduce a number of dropped data packets because of path failure and to keep the continuity of data packet delivery in a presence of faults on main path of data transmission.

The remainder of this paper is organized as follows. In Section 2, we illustrate the communication in WSN. Fault-tolerance issue in WSN is discussed in Section 3. The proposed ENFAT- AODV routing protocol is depicted in Section 4. In Section 5, the results from comprehensive simulations are presented, analyzed and evaluated. Finally, we make the conclusion and future work in Section 6.

2. Communication in Wireless Sensor Network

Further developments in this technology have led to integration of sensors, digital electronics and radio communications into a single integrated circuit (IC) package. Generally wireless sensor network have a base station that communicates through radio connection to other sensor nodes. The required data collected at sensor node is processed, compressed and sent to gateway (sink node) directly or through other sensor nodes.

The sensor nodes are usually scattered in a sensor field as shown in Fig. 1. Each of these scattered sensor nodes has the capabilities to collect data and route data back to the sink. Data are routed back to the sink by a multihop infrastructureless architecture through the sink. The sink may communicate with the task manager node (user) via Internet or satellite as shown in Fig. 1.

3. Fault-Tolerance Issue in Wireless Sensor Network

In WSNs, some sensor nodes may fail or be blocked due to lack of power, physical damage, or environmental interference. The failure of sensor nodes should not affect the overall task of the sensor network. Fault tolerance is the ability of a system to continue providing its specified service despite component failures. It is carried out via fault detection and fault recovery. Since the sensor nodes are prone to failure, WSNs must offer characteristics such as: reliability, availability and fault-tolerance ability.

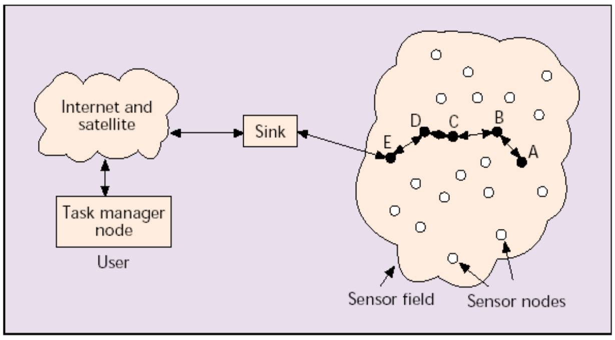


Fig. 1 Wireless sensor nodes scattered in a sensor field [16].

3.1 Source of Faults in Real WSN Applications

Wireless sensor networks are commonly deployed in harsh environment and are subject to faults in several layers of the system [17].

Fig. 2 presents a layered classification of components in a WSN that can suffer faults. A fault in each layer of the system has the possibility to propagate to above levels. For example, a power failure of a node will cause the entire node to fail. If this node is on a routing path, the messages of other nodes relying on this routing path will not be delivered making an entire region of the network silent until the routing path is restored.

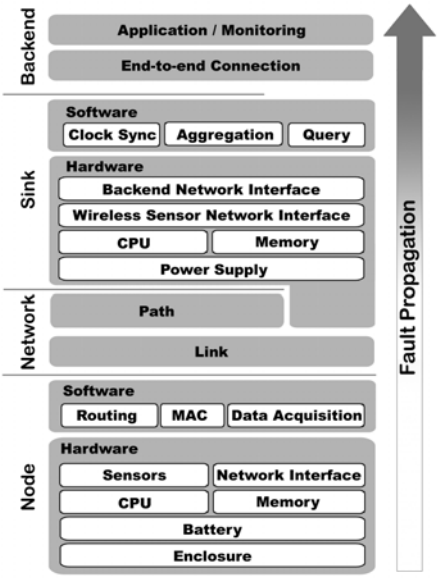


Fig. 2 Fault classification and propagation [17].

In this paper, however, we will concentrate on faults that can happen in the sensor nodes up to the sink.

1) Node Faults: Nodes have several hardware and software components that can produce malfunctions. In [18] due to stress from the environment and inadequate enclosures, the sensor nodes were exposed to direct contact with water causing short circuits. The report of a large-scale deployment in a potatoes field [19] indicated that the antennas from the nodes were quite fragile and would become loose when inserting the node into the packaging.

2) Link Faults: In WSNs, communication links between nodes are highly volatile. In [20], the instability of the links between nodes leads to constant changes in the routing paths.

Radio interference can also cause the link between nodes to become faulty. For instance, in agricultural fields the placement of the nodes must be carefully planned to take into consideration that when plants start growing the link range is considerably reduced, as discussed in [21].

3) Sink Faults: On a higher level of the network, a device (sink) that collects all the data generated in the network and propagates it to the back-end system is also subject to faults of its components.

The sink can be deployed in areas where no permanent power supply is present, in such applications batteries together with solar cells are commonly applied to provide the amount of energy necessary. However, this traditional technique has proven to be inefficient [18]. Although this worked perfectly for other experiments, in the glacial environment the sink suffered a power failure due to snow covering the solar cells for several days.

4. The Enhanced Fault-Tolerant AODV Routing Protocol

In this section, we present the details of our proposed routing protocol operations. Since the purpose of our research is to enhance the fault-tolerant mechanism of AODV routing protocol for WSN, so our protocol description is based on AODV. Furthermore, our modifications to AODV for applying our technique are also introduced.

4.1 ENFAT- AODV Overview

The proposed ENhanced FAult-Tolerant AODV (ENFAT-AODV) routing protocol enables fault-Tolerant, self-starting, multihop routing between participating nodes wishing to establish and maintain a fault-tolerant wireless sensor network. ENFAT-AODV provides quick and efficient route establishment between nodes desiring communication and is designed specifically for ad hoc wireless sensor network which are prone to a lot of failures. Moreover, ENFAT-AODV allows nodes on a main path of data delivery to obtain a backup route, which is used when their main path gets failed, to respond to link breakages in a timely manner. The operation of ENFAT-AODV is loop-free and able to avoid the Bellman-Ford "counting to infinity" problem by exploiting the destination sequence number.

The number of hops along a path is used as a metric for a path selection. If multiple RREPs with same destination sequence number are received by the source, the route with the shortest hop count is chosen.

Route Requests (RREQs) and Route Replies (RREPs) are the same message types as defined by AODV [15]. However, for ENFAT-AODV, we add some fields in the control packets such as “BACKUP” flag (in RREQ and RREP), “UPDATE” flag (in RREQ) and “DistanceToDest” field (in RREQ). These message types are received via UDP, and normal IP header processing applies. Additionally, ENFAT-AODV operation does require certain messages (e.g., RREQ) to be disseminated widely, perhaps throughout the network. The range of dissemination of such RREQs is indicated by the TTL in the IP header. Fragmentation is typically not required.

When a main path of data delivery to the destination (sink node) is needed, the source node will run a “Main Route Discovery” process. During the period of unicasting the main RREP packet back to the source node, each node receiving the main RREP creates backup route towards the destination (runs a “Backup Route Discovery” process) as well. We mostly develop a new fault-tolerance mechanism of AODV routing protocol in this process. Therefore, during data packet delivery period, when the main path gets failure, the node immediately utilizes its backup route to deliver the next coming data packets, instead of the previously broken main route, without an interruption of data packet transmission. As a result, it increases more reliability and availability compared to original AODV routing protocol.

ENFAT-AODV is a routing protocol utilizing also a distance vector algorithm; a node never actually knows a complete path from source to destination, instead, it only knows the direction (which neighbor) to which it should forward a packet in order to reach a given destination. Therefore, it deals with routing table management. Each node in the system contains two separate routing tables called “Main Routing Table” and “Backup Routing Table”. The “Backup Routing Table” is new added from the original AODV for backup route management. Route table information must be kept even for short-lived routes, such as are created to temporarily store reverse paths towards nodes originating RREQs.

Furthermore, ENFAT-AODV also reduces some implementation complexity by eliminating a set of items from the original AODV specifications as follows. First, Hello, RERR (Route Error), and RREP-ACK (Route Reply Acknowledgment) messages are removed to reduce unnecessary control packets in the network. Second, local repair operation is not included in ENFAT-AODV.

4.2 Main Route Discovery

Our technique is incorporated with reactive AODV routing protocol that builds routes on demand via a query and reply procedure. In “Main Route Discovery” Process, ENFAT-AODV does not require any modification to the AODV's RREQ propagation process. When a source node determines that it needs a path to a destination node for transmitting data packets and does not have one available, it broadcasts a main Route Request (main RREQ) packet for that destination node. At each intermediate node, when a main RREQ is received, a reverse route to the source is created. If the receiving node has never received this main RREQ before (by checking the “Flooding ID” and “Source IP Address” in the main RREQ packet), Also it is not the destination node and does not know a “fresh enough” main route to the destination, it will rebroadcast the main RREQ to its neighbors; otherwise, it will silently discard the received packet. If the receiving node is the destination or has a “fresh enough” main route to the destination, it will generate a main Route Reply (main RREP) packet. Then, the main RREP is unicasted in a hop-by-hop fashion to the source. As the main RREP is forwarded back to the source, every intermediate node which processes the main RREP creates a forward route (main route) to the destination. When the source receives the main RREP, it records the main route to the destination in its main routing table. As a result, the main path of data transmission from the source to the destination node is established and ready to use for sending data packets. If multiple RREPs are received by the source, the route with the shortest hop count is chosen. In ENFAT-AODV, the hop count is used to determine the best route.

4.3 Backup Route Construction

The backup routes to the destination for nodes on the main path are established during the phase of forwarding back the main RREP message to the source. We mostly modify the AODV protocol for the fault-tolerance mechanism in this procedure.

During main route reply phase, nodes on a main path (including a source node) which receive a main RREP create a backup route towards a destination node (run a “Backup Route Discovery” process) by broadcasting a route request packet with “Backup flag” set (backup RREQ). The TTL of the packet is initially equal the number of hops along the main path from the backup route requesting node to the destination incremented by one to enable control over how far the backup RREQ is disseminated and to prevent unnecessary network-wide floods of backup RREQs. After broadcasting the backup RREQ, the node waits for a route reply packet with “Backup flag” set (Backup RREP) from the destination itself or an intermediate node which has only active short backup route information for the destination in its backup routing table and is not a node on the main path of data delivery from the source to the destination. It implies that not only the destination can generate the backup RREP but also the immediate node can be responsible this task to reduce the number of producing more control overhead (e.g. backup RREQ) as well as to decrease the backup path establishment time. In ENFAT-AODV, the hop count is used to determine the excellent backup route as well.

For another condition, if a backup RREQ reaches at a node which is on the main path of data delivery from the source to the destination, except the destination node, it discards the received backup RREQ silently. As for the destination, if it receives a backup RREQ directly from the backup route requesting node which is the destination node’s next hop along the main path towards the source node, it also discards the received backup RREQ. The reason behind these conditions is to prevent unnecessarily wide dissemination of the backup RREQ and establishment of useless backup route (overlapping with the main path). The rest of this subsection describes actions taken for backup RREQs that are not discarded.

When the backup RREQ reaches an intermediate node which can directly reply the required backup route information, it first checks the “DistanceToDest” field in the backup RREQ indicating the hop count from the backup route requesting node to the destination along the main path. If the number of hops along the active backup path from the intermediate node to the destination is greater than or equal the “DistanceToDest” field of the backup RREQ, it discards the received backup RREQ silently to prevent creating a too long backup path for the requesting node; Otherwise, it generates a backup RREP packet and unicasts it back to the backup route requesting node along the same path as the backup RREQ was transmitted.

When a backup RREQ arrives at a node which is on the main path from the source to the destination, except the destination node, it discards the received backup RREQ silently to prevent an unnecessarily wide dissemination of the backup RREQ.

For the destination node, if it receives a backup RREQ directly from a backup route requesting node which is its next hop along the main path towards the source, it also silently discards the received backup RREQ to prevent an establishment of useless backup path overlapping with the main path; otherwise, it generates a backup RREP and forwards it back towards the node requesting the backup route.

Once the backup route requesting node receives the expected backup RREP, the backup path from the node to the destination node is established and ready to use.

As shown in Fig. 3, after node11 receives the main RREP from the destination (node16), the main forward route for data packet delivery from node11 to the destination (node16) is established and it subsequently creates its backup route to the destination by generating and broadcasting a RREQ with “Backup” flag set (backup RREQ) with a starting TTL value. When node16 (the destination) receives the backup RREQ, originated by node11, from node12, it generates a backup RREP and unicasts it back towards node11 along the same path as the backup RREQ was transmitted. Furthermore, node16 discards the backup RREQ received directly from node11 (the backup route requesting node) which is its next hop along the main path towards the source node to prevent creating a useless backup route overlapping its main forward route. In addition, node6, which is on the main path of data delivery from the source to the destination node, discards the backup RREQ originated by node11 to avoid a needlessly wide propagation of the backup RREQ. Once node11 receives the expected backup RREP generated by the destination node, the backup path from node11 to the destination node (node16) is established (11->12->16).

In Fig. 4, similarly, after node6 receives the main RREP, generated by node16 (the destination node), from node11, the main forward route for data packet delivery from node6 to the next hop (node11) towards the destination (node16) is established and it subsequently creates its backup route to the destination by generating and broadcasting a backup RREQ with ‘Backup’ flag set with a initial TTL value. Afterwards, when the backup RREQ reaches node12 (an intermediate node) which can satisfy the specified conditions as follows:

- it has an active backup route information entry for the destination,
- it is not a node on the main path,
- and the number of hops from the node itself (node12) to the destination (node16) along the backup path (which is equal one) is less than the “DistanceToDest” value in the backup RREQ (indicating the hop count from node6 (the backup route requesting node) to the destination (node16) along the main path) (which is equal two) to guarantee that it will provide a short backup path to node6,

then it generates a backup RREP and unicasts it back towards node6 along the same path as the backup RREQ was transmitted. In addition, node11 and node1, which are on the main path of data delivery from the source to the destination node, discard the backup RREQ originated by node6 to avoid unnecessarily wide dissemination of the backup RREQ (reduce control overhead). Once node6 receives the expected backup RREP generated by the intermediate node (node12), the backup path from node6 to the destination node (node16) is established (6->7->12->16).

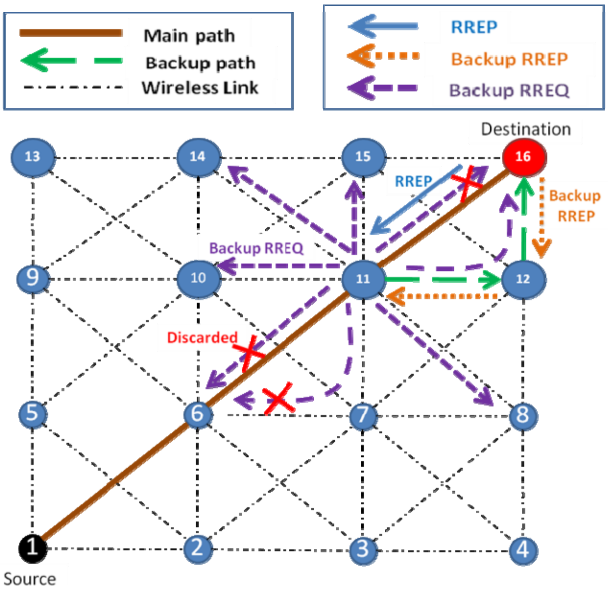


Fig. 3 Backup route establishment: The destination node generates a backup RREP.

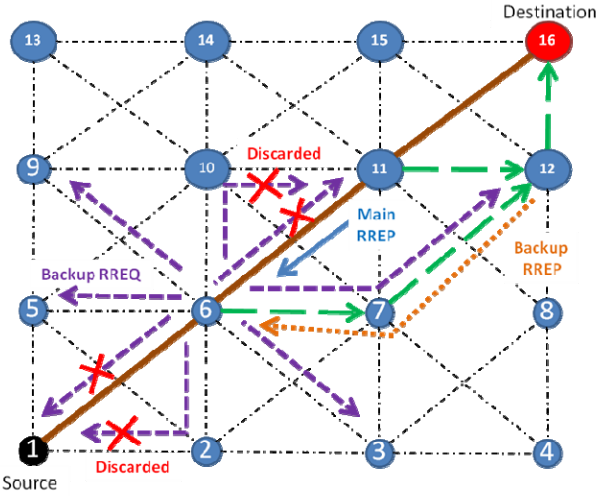


Fig. 4 Backup route establishment: An intermediate node generates a backup RREP.

4.4 Route Maintenance

As data flows from the source to the destination, each node on the main path updates the timers associated with the main route to the source and destination in its main routing table. If a route is not used for some period of time, a node cannot be sure whether the route is still valid; consequently, the node removes the route from its routing table.

Every time when a node forwards a data packet through the main route successfully, it checks whether it has active backup route to the destination. If such active backup route cannot be found, it runs “Backup Route Discovery” process again to obtain a backup route to the destination. In case it can find an active backup routing table entry for the destination in its backup routing table, it will check the expiry time of the active backup routing table entry, if the entry lifetime is almost expired, the node will update its backup route by unicasting a small massage to the destination through its backup route before its backup route will be inactive and wait for a reply. The reason behind such the actions is to ensure that the nodes on the main path always possess a backup route during the data delivery period.

However, data packets are delivered through the primary path unless there is a route disconnection. When a node on the main path, except the destination node, detects a link break for the next hop of the main path while transmitting data (e.g. receives a link layer feedback signal from MAC protocol) or gets a data packet destined to the destination node for which it does not have an active main route, it immediately switches the route of data packet delivery by utilizing its backup route to become a new main forward route (without generating a route error (RERR) message to inform its neighbors) and then forwards the data packet and the coming next through it without an interruption of data transmission. Subsequently, the node on the new main path, which now lacks a backup route, runs a “Backup Route Discovery” process to find a new backup route. Applying the backup path scheme is likely to increase a number of data packets that are able to be delivered to the destinations, since next data packets will not be dropped due to no route for data packet transmission (because of the broken main route) as compared to original AODV.

As shown in Fig. 5, during the data packet delivery period, at node6, after it forwards the 1st data packet to the next hop (node11) towards the destination (node16), it detects a failure of the link between the node itself and node11 because node11 gets failed. As a result, the 1st data packet is lost or dropped.

As shown in Fig. 6, after node6 detects the main forward route failure, it instantaneously switches the route of data delivery by employing its backup route (the route from node6 to node7) to become its new main forward route. Afterwards, when node6 gets the next coming data packet from node1 (the source), it immediately forwards the data packet through the new main path (6->7->12->16) without any interruption of data transmission. Consequently, the new main path of data delivery from the source (node1) to the destination (node16) becomes 1->6->7->12->16. Subsequently, nodes on the new main path, except the destination node, which now lack a backup path (node6, node7 and node12) run “Backup Route Discovery” process to get their new backup route.

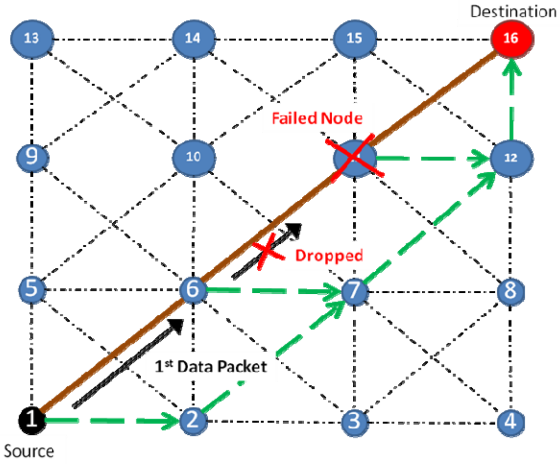


Fig. 5 Route maintenance: Node6 detects a main route failure.

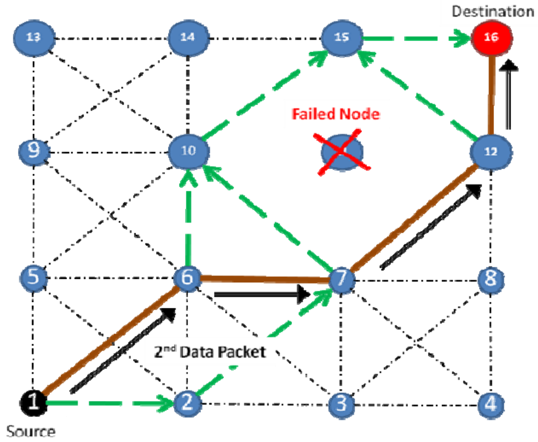


Fig. 6 Route maintenance: Node6 switches the route of data packet delivery.

5. Simulation Experiments

In this section, we use the QualNet 4.5.1 Simulator [22] to simulate the ENFAT-AODV routing protocol, and compare the performance of ENFAT-AODV with original AODV. According to the simulation results evaluation, it shows the ENFAT-AODV is greatly suitable for high failure rate WSN.

5.1 Simulation Environment

To evaluate the performance improvements made by ENFAT-AODV, we compare the simulation results of AODV protocol with and without applying our proposed fault-tolerance mechanism.

In the simulation model, there are 100 wireless sensor nodes (MICAz) deployed in a 3500x3500 m^2 field, the simulation time is set to 100 seconds. We set all the nodes are static (no movement). The type of the wireless propagation model is Free Space Propagation [23]. The maximum radio propagation range is set to 550 m. Each node sets the transmission power and the receiver sensitivity to 15 and -89 dBm respectively. The type of antenna model is omni-directional with a height of 1.5 m and 0 dB antenna gain. The source node generates constant bit rate (CBR) data streams with packet interval of 0.05 second. The size of data payload is 512 bytes. The link bandwidth and channel frequency is set to 2 Mbps and 2.4 GHz respectively. All sensor nodes communicate each other by using wireless multihop communication. Table 1 summarizes the simulation parameters.

Table 1: Simulation Parameters

Parameter	***Values***
Simulation area	3500x3500 m^2
Simulation time	100 seconds
Sensor node type	MICAz
Number of the sensor nodes	100 nodes
Data packet size	512 bytes
Packet interval	0.05 s
Max. propagation range	550 m
Propagation model	Free Space
Link Bandwidth	2 Mbps
Channel frequency	2.4 GHz

Based on the simulation setting, during data transmission, we specify an interface fault permanently on some nodes along the main path of data packet delivery at specified time to make the main path broken. When a node is specified an interface fault, it will get failed to receive and transmit any packets (as the node disappears in the network).

According to the purpose of our simulation, we desire to analyze the effect of increased number of failures on main path (up to six times) upon main system performance metrics such as the throughput, number of dropped data packets, average jitter and control overhead in the network by comparing three routing protocols: ENFAT-AODV, AODV without Local Repair function and AODV with Local Repair function.

5.2 Throughput Analysis

In this simulation, throughput is calculated by using the following formula:

TP (bit/s) = [(Total of (data) bytes rcved by Dest * 8.0) / (T_E – T_F)]
where
TP = Throughput (bit/s).
T_E = the time when the simulation ended (s).
T_F = the time when first data packet received by Destination (s).

Fig. 7 shows the results of the destination node's throughput against the increased number of failures on main path. In comparison, the ENFAT-AODV gives highest throughput of the destination node. However, AODV with Local Repair function provides higher throughput than AODV without Local Repair function. With the increased number of failures on main path, for AODV, the throughput result decreases rapidly because when the path of data delivery gets failed during data transmission period, many data packets are buffered or dropped in the network due to no active route for data delivery that the system spends some time for "Route Discovery" process until it can find a new active route for data transmission; the number of "Route Discovery" process running depends on the number of failures on main path. On the other hand, for ENFAT-AODV, the throughput result decreases much more slowly, even, with high number of failures on main path since the system utilizes the backup route when the main path breaks. The next coming data packets will be immediately delivered through the backup route without any interruption of data delivery.

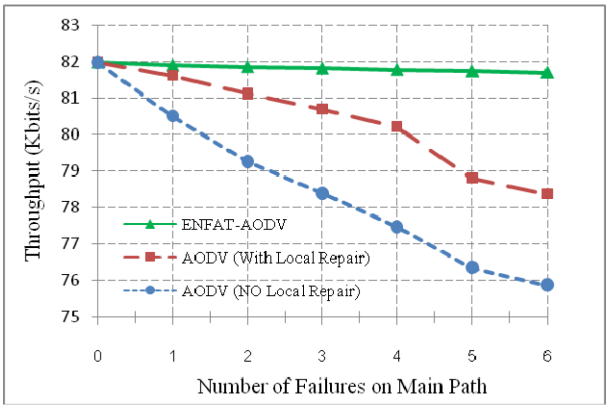


Fig. 7 Throughput with the increased number of failures on main path.

5.3 Analysis of Number of Dropped Data Packets

Fig. 8 shows the results of the number of dropped data packets against the increased number of failures on main path. From the graph, it is observed that ENFAT-AODV gives lowest number of dropped data packets. However, AODV with Local Repair function provides lower number of dropped data packets compared to the results of AODV without Local Repair function. During data transmission period of the system, when the main path of data delivery breaks because of the interface fault, ENFAT-AODV always switches the backup route instantaneously to become the new main path for the delivery of next coming data packets. Therefore, even with high number of failures on main path, it provides much low number of dropped data packets as we expected. On the other hand, with highly increased number of failures on main path, AODV also extremely increases the number of dropped data packets because after the path of data delivery gets failed during data transmission period, it needs to find a new path for the transmission of next coming data packets by running “Route Discovery” process. While the process is being running, the source node still generates and transmits the data packets to the destination through the same path which was broken causing some data packets to be dropped. Moreover, the number of “Route Discovery” process running depends on the number of failures on main path. As a result, with high number of failures on main path, a lot of data packets are dropped in the network.

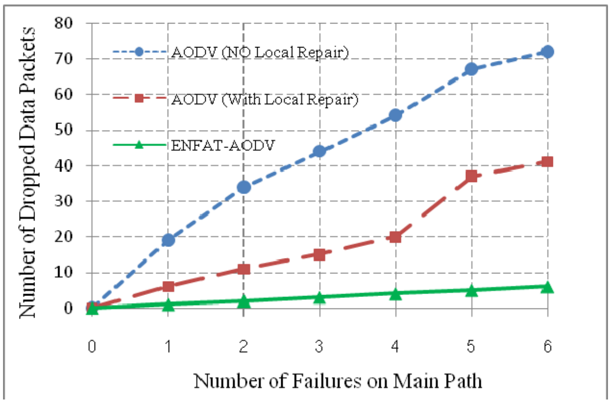


Fig. 8 Number of dropped data packets with the increased number of failures on main path.

5.4 Jitter Analysis

In this simulation, jitter is used as a measure of the variability over time of the data packet latency across a network. Average jitter is calculated by using the following formula:

Avg. Jitter = $\frac{\sum_{i=2}^{n} Jitter(i)}{(n-1)}$ second

Jitter(i) = Jitter(i-1) + [(|D(i-1,i)| - Jitter(i-1))/16] second

where
Jitter(i) is the jitter after the Destination receives an i-th data packet.
D is the difference of relative transit times for the two data packets.
n is total number of data packets received by Destination.

Most of applications for WSNs are real-time applications, typically, involving some kinds of monitoring, tracking, or detecting such as weather monitoring, object tracking, fire detection etc. The average jitter is an important QoS factor in an assessment of network, especially, in a real-time application. A system with low jitter provides good QoS.

Fig. 9 shows the results of average jitter against the increased number of failures on main path. As expected, from the graph, with increased number of failures on main path, it is observed that ENFAT-AODV provides lowest and most stable average jitter because it always utilizes the backup path in case of broken main path without an interruption of data transmission. However, AODV with Local Repair function gives lower average jitter compared to the results of AODV without Local Repair function. When a failure on main path occurs, AODV with Local Repair function can obtain a new main path for the transmission of next data packets faster than AODV without Local Repair function.

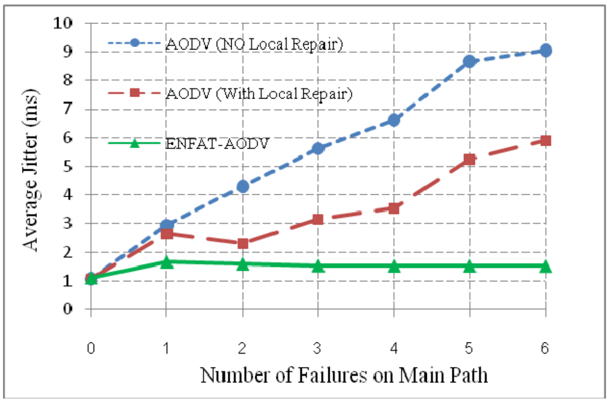


Fig. 9 Average jitter with the increased number of failures on main path.

5.5 Control Overhead Analysis

Based on our simulation, control overhead is measured by using the following formula:

Control Overhead (packet) = Total no. of RREQ, RREP and RERR packets initiated and forwarded in the network.

Fig. 10 shows the results of control overhead against the increased number of failures on main path. According to the simulation results, with increased number of failures on main path, AODV with Local Repair function provides lower control overhead than AODV without Local Repair function. In addition, with no failure of main path of data delivery in the simulation, ENFAT-AODV produces more control overhead than AODV. For ENFAT-AODV, the extra control packets are initiated and forwarded for backup route establishment and updating. However, if the number of failures on main path is highly increased, the control overhead generated for AODV also greatly increases. From the graph, it is observed that ENFAT-AODV gives lower control overhead compared to AODV with Local Repair function if the number of failures on main path is more than two times and compared to AODV without Local Repair function if there is at least one time of failure on main path occurred in the simulation.

In high main path failure rate, AODV runs “Route Discovery” process to find a new route of data delivery and initiates a RERR packet because of route error many times; as a result, it produces high control overhead as well. On the other hand, for ENFAT-AODV, it runs “Main Route Discovery” process just one time when it needs a main path of data delivery towards the destination in start time of the simulation and also runs “Backup Route Discovery” process to obtain backup paths. Afterwards, when the main path breaks during data transmission, the system does not generate RERR packet as in AODV but, instead, it switches the backup route immediately to become the new main path for the delivery of next coming data packets and only some nodes along the new main path which lack for a backup route run “Backup Route Discovery” process to find their new backup route that produces less network control packets compared to “Main Route Discovery” process which is mostly run by the source node.

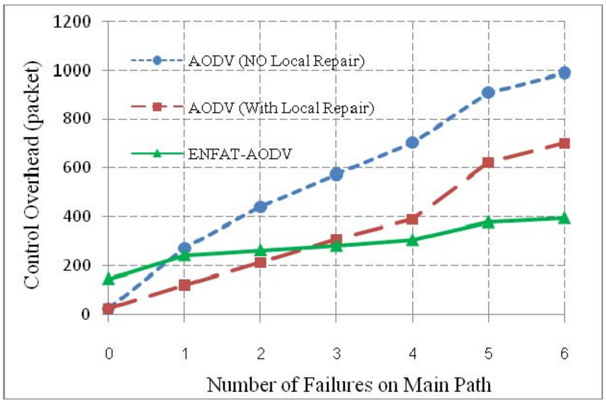


Fig.10 Control overhead with the increased number of failures on main path.

6. Conclusion and Future Work

In this paper, we propose the ENhanced FAult-Tolerant AODV (ENFAT-AODV) routing protocol for transmitting data packet in wireless sensor networks which are prone to various failures. The ENFAT-AODV utilizes the backup route technique to improve reliable data packet delivery and keep the system operations still running continually in presence of some faults (link breaks or node failures). The backup routes are employed only when data packets cannot be delivered through the main path. As a result, the reliability, availability and fault-tolerant ability of the network are achieved.

As a case study, we applied our proposed fault-tolerance mechanism to AODV routing protocol and measured performance improvements. According to the proposed protocol design, ENFAT-AODV can also reduce some implementation complexity by eliminating a set of items from the original AODV specifications as follows. First, Hello, RERR (Route Error), and RREP-ACK (Route Reply Acknowledgment) messages are removed to reduce unnecessary control packets in the network. Second, local repair operation is not included in ENFAT-AODV.

The developed scenarios have been simulated using QualNet 4.5.1 which is an efficient network simulator. The simulation results indicate that the proposed technique provides robustness to data packet delivery for high failure rate WSN and enhances protocol performance. With the greatly increased number of failures on data delivery path, the ENFAT-AODV can improve the throughput, decrease the number of dropped data packets, reduce the average jitter, and provide low control overhead in the network. With no failure of main path of data delivery in the scenario, although, the ENFAT-AODV produces extra control packets because of backup route establishment and updating, it provides a little bit more network energy consumption as compared to the results of original AODV. However, the ENFAT-AODV performs well only in the static or very low movement scene.

In a strong WSN (no failure occurred on a main path of data delivery), AODV is more suitable than ENFAT-AODV because the backup route may be useless; the data packets are delivered through the same main path without utilizing a backup route. However, ENFAT-AODV is appropriate to be deployed in wireless sensor networks, especially for high failure rate systems, which are prone to a lot of failures.

For future work, we plan to further evaluate our proposed protocol by using more detailed and realistic channel models with fading and obstacles in the simulation. Moreover, we also plan to further improve the system limitation and drawback as much as possible. Most importantly, we strongly believe the advantage of providing an efficient fault-tolerance mechanism to the WSN will be greatly beneficial in that environment.

Acknowledgment

This work (research) is financially supported by the Research Management Center of International Islamic University Malaysia (IIUM) through the IIUM Research Endowment Fund.

References

[1] T. Haenselmann, “Sensornetworks”. http://www.informatik.uni-mannheim.de/~haensel/sn_book. Retrieved in May 2010.

[2] J. N. Al-Karaki and A. E. Kamal, “Routing techniques in wireless sensor networks: A survey”, IEEE Wireless Communications Magazine, Vol. 11, pp. 6 – 28, 2004.

[3] W. Xiao, M. Xu and Y. Chen, “A Self-adaptive Fault-Tolerant Mechanism in Wireless Sensor Networks”, In M. Peter, C. J. Nong and W. C. Li (Eds.), Scalable Information Systems (pp. 228– 240), Vol. 18, Springer, 2009.

[4] M. Demirbas, “Scalable Design of Fault-Tolerance for Wireless Sensor Networks”, Doctoral Dissertation. Ohio State University, USA, 2004.

[5] G. Gupta and M. Younis, “Fault-Tolerant Clustering of Wireless Sensor Networks”, IEEE Wireless Communications and Networking Conference (WCNC), New Orleans, Louisiana. 2003.

[6] L. B. Ruiz, I. G.. Siqueira, L. B. e-Oliveira, H. C. Wong et al., “Fault Management in Event-driven Wireless Sensor Networks”, 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems (MSWiM), Venice, Italy, 2004.

[7] S. Harte and A. Rahman, “Fault Tolerance in Sensor Networks Using Self-Diagnosing Sensor Nodes”, IEEE International Workshop on Intelligent Environment, Colchester, UK, 2005.

[8] I. Chakeres and L. K. Berndt, “AODVjr, AODV Simplified”, 3rd ACM international symposium on Mobile ad hoc networking & computing (MobiHoc), Lausanne, Switzerland, 2002.

[9] J. Chhabra, “Readme for TinyAODV Stack”, 2003. http://tinyos.cvs.sourceforge.net/viewvc/tinyos/tinyos-1.x /contrib/hsn/README_TinyAODV?revision=1.1.1.1 &view=markup. Retrieved in May 2010.

[10] C. E. Perkins, E. B. Royer and I. Chakeres, “Ad hoc On-Demand Distance Vector (AODV) Routing”, Internet draft. The Internet Engineering Task Force (IETF), 2004. http://moment.cs.ucsb.edu/pub/draft-perkins-manet-aodvbis-01.txt. Retrieved in May 2010.

[11] G. Montenegro and N. Kushalnagar, “AODV for IEEE 802.15.4 Networks”, Internet draft. The Internet Engineering Task Force (IETF), 2005. http://potaroo.net/ietf/idref/draft-montenegro-lowpan-aodv/ Retrieved in May 2010.

[12] K. Kim, S. D. Park, G. Montenegro, S. Yoo and N. Kushalnagar, “6LoWPAN Ad Hoc On-Demand Distance Vector Routing (LOAD)”, Internet draft. The Internet Engineering Task Force (IETF), 2006. http://tools.ietf.org/id/draft-daniel-6lowpan-load-adhoc-routing-02.txt. Retrieved in May 2010.

[13] C. Gomez, P. Salvatella, O. Alonso and J. Paradells, “Adapting AODV for IEEE 802.15.4 mesh sensor networks: theoretical discussion and performance evaluation in a real environment”, IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Buffalo, New York, 2006.

[14] W. Li, M. Chen and M. M. Li, “An Enhanced AODV Route Protocol Applying in the Wireless Sensor Networks”, In B. Y. Cao (Ed.), Fuzzy Information and Engineering V.2 (pp. 1591-1600), Vol. 62. Heidelberg: Springer, 2009.

[15] C. E. Perkins, E. B. Royer and S. Das, “Ad hoc On Demand Distance Vector Routing (AODV)”, Request for Comments (RFC) 3561. The Internet Engineering Task Force (IETF), 2003.

[16] I. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, “A survey on sensor networks”, IEEE Communications Magazine, 40 (8), 102-114, 2002.

[17] L. M. S. D. Souza, H. Vogt and M. Beigl, “A survey on fault tolerance in wireless sensor networks”, SAP Research. Karlsruhe University, Germany, 2007.

[18] K. Martinez, P. Padhy, A. Riddoch, H. Ong and J. Hart, “Glacial environment monitoring using sensor networks”, Workshop on Real-World Wireless Sensor Networks (RealWSN), Stockholm, Sweden, 2005.

[19] K. Langendoen, A. Baggio and O. Visser, “Murphy loves potatoes: experiences from a pilot sensor network deployment in precision agriculture”, 20th IEEE International Parallel and Distributed Processing Symposium (IPDPS), Rhodes Island, Greece, 2006.

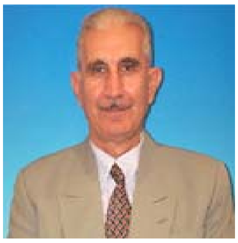
[20] T. Schmid, H. D. Ferri`ere and M. Vetterli, “Sensorscope: Experiences with a wireless building monitoring sensor network”, Workshop on Real-World Wireless Sensor Networks (RealWSN), Stockholm, Sweden, 2005.

[21] J. Thelen, D. Goense and K. Langendoen, “Radio wave propagation in potato fields”, 1st IEEE International Workshop on Wireless Network Measurement (WiNMee), Lago di Gardi , Italy, 2005.

[22] Scalable Network Technologies, Inc., “QualNet Simulator”, 2006. http://www.scalable-networks.com/products/qualnet/. Retrieved in May 2010.

[23] Eltahir, I.K., "The Impact of Different Radio Propagation Models for Mobile Ad hoc NETworks (MANET) in Urban Area Environment”, Wireless Broadband and Ultra Wideband Communications, pp. 30-30, August 2007.

Zamree Che-Aron received the B.Eng. degrees in Computer Engineering, from Prince of Songkhla University in 2007. Currently he is pursuing the M.S. degree in Computer and Information Engineering at International Islamic University Malaysia. His research interests are in the areas of Wireless Networks, Mobile Communications, and Broadband Communications.



Dr. Wajdi Fawzi Al-Khateeb received his PhD from the International Islamic University, Malaysia and his MSc from the Technical University of Berlin, Germany. His research interest is mainly in the Reliability Engineering, Fault Tolerant Systems, QoS Networking, Microwave Radio Links. He is currently an assistant Professor in the department of Electrical and Computer Engineering, International Islamic University Malaysia.

Dr. Farhat Anwar received a PhD degree in Electronic and Electrical Engineering from the University of Strathclyde UK in 1996 and his MSc from the University of Dhaka, Bangladesh. His research interest includes QoS in IP networks, routing in Ah-hoc and sensor networks, computer simulation and performance analysis, and biometrics. He has published extensively in international journals and conferences. He has been with IIUM since 1999 and currently working as a Professor in the department of Electrical and Computer Engineering.