

Fault Tolerant Multi-Criteria Multi-Path Routing in Wireless Sensor Networks

Koffka Khan

Department of Computing and Information Technology, The University of the West Indies, Trinidad and Tobago, W.I
E-mail: koffka.khan@sta.uwi.edu

Wayne Goodridge

Department of Computing and Information Technology, The University of the West Indies, Trinidad and Tobago, W.I
E-mail: wayne.goodridge@sta.uwi.edu

Abstract— The Ad Hoc On-Demand Multi-Path Distance Vector (AOMDV) routing protocol allows the transport of data along one or more paths in wireless sensor networks (WSNs). The path chosen is based on a single shortest path hop count metric. The data on some WSNs is mission critical, for example, military and health care applications. Hence, fault tolerance in WSNs is becoming increasingly important. To improve the fault tolerance of WSNs in lossy environments, this work adds to the AOMDV routing protocol as it incorporates an additional packet loss metric. This Multi-criteria AOMDV or M-AOMDV is evaluated using the ns2 simulator. Simulations show that M-AOMDV maintains relatively low packet loss rates when the WSN is experiencing loss.

Index Terms— Multipath; Multi-criteria; AOMDV; WSNs; M-AOMDV; Fault Tolerance

I. INTRODUCTION

A WSN [37],[18],[28] has to be able to tolerate network failures which threaten to disrupt its services. For example, in WSNs located at military hotspots, where small packet delays can be detrimental to the sighting of specific enemy targets. The data gathered by sensor nodes is usually sent to one base station or sink node. This communication model is called many-to-one (M-1). Packet loss near sink nodes can be particularly harmful to WSNs because of the M-1 communication model and the convergent nature of upstream traffic in WSNs means that there usually is a high traffic flow rate near the sink. This means that WSNs are more susceptible to packet loss in these regions with severe service disruption to applications.

Multipath routing [19],[12],[24] can provide fault tolerance for wireless networks [22]. To provide fault tolerance at least two paths are needed to transfer packets to the destination node. Although, fault tolerance can be achieved by sending packets on multiple paths to the destination node, it can also be achieved if there is a fault discovery mechanism which allows nodes to switch to more reliable paths when faults are detected on the main path. Thus multipath routing protocols, for example, AOMDV, can provide advantages for the wireless environment provided that such a fault discovery mechanism exists.

Recently, routing problems have become increasingly important given the emergence of applications that require guarantees on a range of Quality of Service (QoS) parameters such as delay, cost, bandwidth, loss rate, jitter, etc. In many cases the QoS was measured against route packet dropping rates and so in that context would be a measure of fault tolerance of the network. These new challenges have led to multi-criteria routing problem [8],[30],[10],[11]. Several algorithms are proposed to resolve multi-criteria routing problems. Some of these algorithms consider two main metrics: cost and delay [38]. Two approaches exist for these algorithms. A first approach considers the multi-criteria routing problem as a mono-objective optimization one that minimizes the cost under the delay constraint. A second approach uses a multi-objective formulation: to solve this class of problems, some proposed works use meta-heuristics such as genetic algorithms [14], taboo search [33], or ant colonies [21]. Other works propose exact multi-objective algorithms. The most efficient one is SAMCRA which was proposed by Van Mieghem et al. in [34]. SAMCRA is an exact multi-criteria routing algorithm. However, the major drawback of SAMCRA is its complexity [6].

The Simple Additive Weights (SAW) [40] is a multi-criteria decision making (MCDM) technique which allows a decision maker to select the best alternative from a list of alternatives given that the selection process uses multiple criteria. Therefore, the SAW method can select the most appropriate path based on a tradeoff among two or more metrics. The principle of choosing paths based on multiple metrics may prove beneficial to WSN performance.

Fault tolerance is the ability of a system to deliver a desired level of functionality in the presence of faults [9]. Since the sensor nodes are prone to failure, fault tolerance should be seriously considered in many sensor network applications. Actually, extensive work has been done on fault tolerance and it has been one of the most important topics in WSNs [17]. Fault recovery procedures should allow the network to maintain its' operational levels.

This paper proposes an improvement to the AOMDV protocol called the Multi-metric Ad hoc On-demand Multi-path Distance Vector (M-AOMDV) routing protocol, which attempts to select paths with lower packet

loss rates in the presence of packet loss. M-AOMDV acts as a fault tolerant defense mechanism as it works proactively to ensure the continuance of minimal service levels to WSN applications at the sink mote. Hence, the main purpose of this paper is to answer the question: Does path selection based on both hop count and packet loss help to improve WSN fault tolerance [35],[4],[31]. We attempt to evaluate the fault tolerance of WSN, using the M-AOMDV routing protocol.

We present background and related work in Section II. The M-AOMDV routing protocol is presented in Section III. Experimental design is outlined in Section IV with metrics and experimental results. Further, in Section V the experimental results are given together with a discussion. Finally, this paper closes off with the conclusion in section VI.

II. BACKGROUND AND RELATED WORK

In Section A multipath routing in WSNs is described. Then, in section B the AODV routing protocol is described. Further, in section C the AOMDV routing protocol is illustrated. Finally, in section D the SAW MCDM method is explained.

A. Multi-Path Routing in WSNs

The AOMDV [1] is a multipath routing protocol which is an extension to the Ad Hoc On-Demand Distance Vector (AODV) routing protocol. Unlike AODV, every RREP is being considered by the source mote and thus multiple paths are discovered in one route discovery. The intermediate motes maintain multiple path entries in their respective routing table. The route entry table at each mote also contains a list of next hop along with the corresponding hop counts. Every mote maintains an advertised hop count for the destination. Route advertisements of the destination are sent using this hop count. The advantage of using this approach is that when a network error occurs there is no need to call a new route discovery process. The source mote selects a next best path in terms of hop count from its routing table to continue the communications. The AOMDV routing protocol uses a shortest path selection mechanism based on hop count as the single routing metric. The algorithm for producing paths in AOMDV produces both node and link disjoint paths. Node disjoint paths form distinct subsets within the WSN, such that no two paths contain the same mote, while link disjoint paths can have motes in common. Compared to other on demand multipath protocols AOMDV is unique as there is no high inter-nodal coordination overhead like TORA [5] and ROAM [1].

B. The AODV Routing Protocol

AODV is a pure on-demand routing protocol where creation of routes happens only when desired by the source node. A unicast route is a route from a source node to a destination node. AODV has two phases, (1) Route establishment or discovery and (2) Route maintenance. A node does not perform route discovery or

maintenance until it needs a route to another node or it offers its services as an intermediate node. The Route Discovery Process is completed when a route is found and all possible routes have been examined. The AODV routing protocol uses a broadcast route discovery mechanism with hop-by-hop routing from each network mote to the next [6],[16],[27]. Sequence numbers are assigned to routes and routing table entries to supersede stale cached routing entries. Every node maintains two counters, the node sequence number and broadcast ID.

Route Request (RREQ) messages are generated when node S wants to send a message to node D (see Fig. 1). It shows the RREQ packet is on its' last hop transmission to reach the sink or destination mote. It is important to note that a reverse path is formed on the transmission of each RREQ along a hop. This is shown by the dotted arrow from mote T to mote S and from mote V to mote T. Eventually when the RREQ packet reaches the sink mote a complete path would be known by the sink mote to the source mote. Therefore after the final hop the complete forward path will be known by the sink mote, D.

S searches its route table for a route to D. If there is no route, S initiates a RREQ message with the following components (1) IP addresses of S and D, (2) current sequence number of S and the last known sequence number of D, (3) broadcast ID from S (broadcast ID is incremented each time S sends a RREQ message). The <broadcast ID, IP address> pair of the source S forms a unique identifier for the RREQ. Suppose a node P receives the RREQ from S. P first checks whether it has received this RREQ before. Each node stores the <broadcast ID, IPaddress> pairs for all the recent RREQs it has received.

If P has seen this RREQ from S already, P discards the RREQ. Otherwise, P processes the RREQ where (1) P sets up a reverse route entry in its route table for the source S and (2) this entry contains the IP address, current sequence number of S, number of hops to S and the address of the neighbour from whom P got the RREQ.

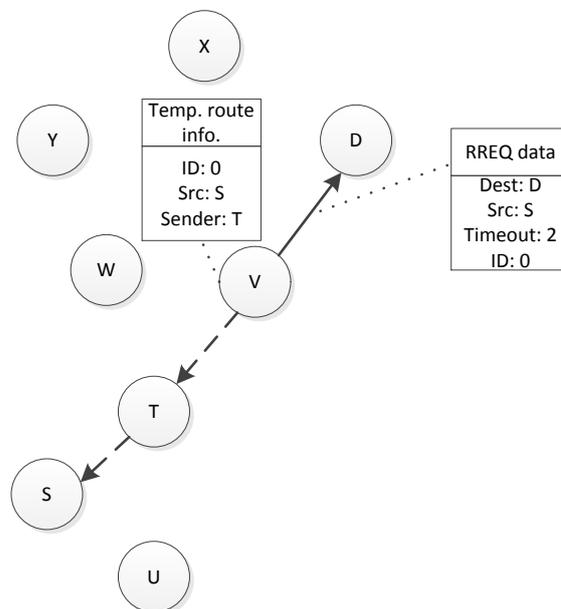


Fig. 1. A RREQ Packet: Formation of a Forward Path

Route Reply (RREP) messages are generated by the destination node in response to the first RREQ (see Fig. 2). At this stage in the route discovery process a single forward path is formed, which is indicated by the dashed arrow from mote V to D and T to V. On the final hop the complete forward path will be known by the source mote, S. The destination then unicasts this RREP to the source node with the <source, destination> pairs reversed. It is now the source node.

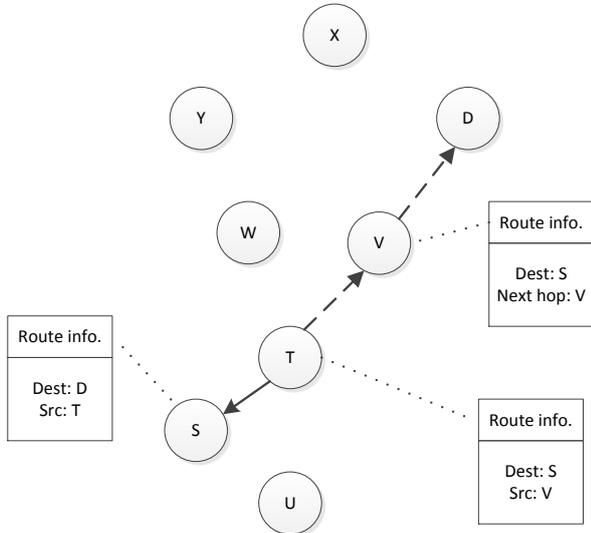


Fig. 2. A RREP Packet: Formation of a Backward Path

An intermediate node P may receive more than one RREP for the same RREQ. P forwards the first RREP it receives and forwards a second RREP later only if (1) the later RREP contains a greater sequence number for the destination, or (2) the hop-count to the destination is smaller in the later RREP. Otherwise, it does not forward the later RREPs. This reduces the number of RREPs propagating towards the source. RREP eventually makes it to the source (see Fig. 2), which can use the neighbor sending the RREP as its next hop for sending to the destination.

If a link breaks down the intermediate node tries to perform a local repair to the needed destinations. Also a Route Error message is sent to upstream neighbors, which lists all the destinations which are now unreachable and a “DestCount” field is used to indicate the number of unreachable destinations. When a node receives a RERR it has several possible actions to perform, (1) it checks whether the sender is its next hop towards the destination, (2) deletes or invalidates the route towards the destination if needed, (3) forwards the RERR upstream if needed or (4) rediscovers route if still needed.

Route changes can be detected by (1) failure of periodic HELLO packets, (2) failure or disconnect indication from the link level or (3) failure of transmission of a packet to the next hop (can detect by listening for the retransmission if it is not the final destination). The upstream (toward the source) node detecting a failure propagates a route error (RERR) packet to the source node with a new destination sequence number and a hop count of infinity (unreachable).

The source (or another node on the path) can rebuild a path by sending a RREQ packet. Maintenance of a Route is needed only until the destination becomes inaccessible along every path from the source or until the route is no longer desired. Nodes that are not on active paths do not maintain routing information and do not participate in routing table exchanges. Routes are based on dynamic table entries maintained at intermediate nodes. Local HELLO messages are used to determine local connectivity, which can reduce response time to routing requests and trigger updates when necessary. Once a unicast route has been established between two nodes S and D, it is maintained as long as S (source node) needs the route. If S moves during an active session, it can reinitiate route discovery to establish a new route to D. When D or an intermediate node moves, a route error (RERR) message is sent to S.

Advantages of AODV protocol include no central administrative system to handle routing, high scalability, need for broadcast is minimized, reduced control messages, quickly reacts to changes in the network, quick response to link breakage in active routes, loop free routes, prevents network flooding during discovery and repairs breaks in active routes locally instead of notifying source. However, one of the major disadvantages of AODV is the high latency due to the route discovery only being reactive.

C. The AOMDV Routing Protocol

Ad Hoc On-Demand Multipath Distance Vector (AOMDV) routing protocol [20] is an extension of the AODV routing protocol with the addition of enabling multiple paths to be found between a given source and destination node. Two of the major goals of AOMDV were to improve on the AODV protocol by answering the following questions: (1) In the AODV framework, how to compute multiple paths between source and destination during route discovery? and (2) How to do this with minimal additional overhead to the AODV framework? Like AODV, AOMDV has an on-demand flood based route discovery mechanism, uses a distance vector routing algorithm and hop-by-hop routing (routing list is sorted based on hop count). There are route discovery and maintenance phases like AODV, but there are multiple paths per route discovery. The protocol ensures loop free paths similar to that of AODV, but an additional feature is the assurance that all paths found are disjoint. In the route maintenance phase AOMDV uses alternate routes on a route failure. New route discovery is only needed when all routes fail. This will result in a fewer number of ‘overall’ route discoveries and an advantage of having a reduction in delay and routing overhead for a given time segment. Major uses of AOMDV would be in MANETs (Mobile Ad hoc NETWORKs) [29], similarly to AODV and more recently in VANETs (Vehicular Ad hoc NETWORKs) [2]. One drawback with AOMDV may be the fact that we are not sure if alternate path works.

Compared to other on demand multipath protocols AOMDV is unique as there is no high inter-nodal coordination overhead like Temporally-Ordered Routing Algorithm, TORA [25], alternative paths are disjoint,

there is no use of source routing and minimum overhead is used to get alternative paths when compared to AODV with reuse of alternate path routing information. Loop freedom is enforced by the use of sequence numbers. Every node maintains a monotonically increasing sequence number for itself and separately maintains the highest known sequence numbers for each destination in the routing table (called “destination sequence numbers”). Destination sequence numbers are tagged on all routing messages, thus providing a mechanism to determine the relative freshness of two pieces of routing information generated by two different nodes for the same destination. The AOMDV protocol maintains an invariant, similarly to AODV that destination sequence numbers monotonically increase along a valid route, thus preventing routing loops. During route discovery a node can receive a routing update via a neighboring mote. The routing table structure of AODV (see Fig. 3) contains the following fields: (destination, sequence number, hop count, next hop, timeout), while the AOMDV routing protocol contains a route list for each destination, which contains an additional last hop field: AOMDV (destination, sequence number, advertised hop count, route list(next hop_m, ..., next hop_n, last hop_m, ..., last hop_n, hop count_m, ..., hop count_n, timeout_m, ..., timeout_n).

Route Request (RREQ) or Route Reply (RREP) packet either forms or updates a reverse or forward path. In AOMDV these routing discovery messages via a RREQ or RREP are referred to as “route advertisements.” During route discovery the source mote that requires a path to a sink mote broadcasts a RREQ packet. This RREQ packet may reach the destination mote by traversing a sequence of multi-hops over one or more motes. At each wireless hop towards the destination the AOMDV routing protocol forms a series of backward pointers or paths which will form a reverse path (see Fig. 2) to the source if the destination mote is reached. Indeed, the main result is the formation of reverse paths from the sink mote to the source mote. This reverse path formation is similar to the path left using bread crumbs in the Hansel and Gretel Fairytale [3]. If there are no intermediate motes with a path to the sink mote then the packet will eventually reach the sink mote. The sink mote in response to this RREQ packet will send a new RREP packet with the destination mote being the source mote. At each hop along the path back towards the source mote the forward path to the destination is recorded. Indeed, the main result is the formation of forward paths from the source mote to the sink mote.

AOMDV replaces the hop count variable with advertised hop count and for each route to a particular destination AOMDV stores a route list. The next hop variable is kept but for each route there may be multiple next hops or neighbors. Similarly as with AODV hop count and timeout variables are kept for each route. The added field in AOMDV which may be repeated multiple times with different values is the last hop to the destination node. As was stated this is used in the computation of multiple alternate disjoint routes to the destination.

D. Simple Additive Weights (SAW)

In a MCDM problem [7] there are four components, namely: (1) alternatives, (2) criteria, (3) relative importance (weights) of each criterion, and (4) criterion values for each alternative. A decision table [15] consisting of these four components is shown in Table 1. The decision table shows alternatives, A_i ($1 \leq i \leq n$), criteria, C_j ($1 \leq j \leq m$), weights of criteria, w_j ($1 \leq j \leq m$) and the measures of performance of alternatives, x_{ij} . The SAW MCDM method is very simple and popular. The decision matrix given in Table 1 is normalized by equations (1) and (2):

Table 1. Components of a Decision Problem

| | Weights | w_1 | w_2 | w_3 | ... | w_m |
|--------------|----------|----------|----------|----------|-----|----------|
| | Criteria | C_1 | C_2 | C_3 | ... | C_m |
| Alternatives | | | | | | |
| A_1 | | x_{11} | x_{12} | x_{13} | ... | x_{1m} |
| A_2 | | x_{21} | x_{22} | x_{23} | ... | x_{2m} |
| A_3 | | x_{31} | x_{32} | x_{33} | ... | x_{34} |
| ... | | ... | ... | ... | ... | ... |
| A_n | | x_{n1} | x_{n2} | x_{n3} | ... | x_{nm} |

$$z_{ij} = \frac{(x_{ij} - b_j)}{(a_j - b_j)}, i = 1, \dots, n; j \in \Omega_b \quad (1)$$

$$z_{ij} = \frac{(a_j - x_{ij})}{(a_j - b_j)}, i = 1, \dots, n; j \in \Omega_c \quad (2)$$

where z_{ij} are normalized criterion values, a_j is the $\max(x_{ij})$ for criterion j and, b_j is the $\min(x_{ij})$ for criterion j , Ω_b and Ω_c are, respectively, the sets of benefit and cost criteria. Associated with each criterion column j in the decision matrix z_{ij} is weighting w_j such that the sum of the sums equals 1. The overall assessment of each alternative is computed by equation (3). The greater the value y_i means that the alternative is ranked higher.

$$y_i = \sum_{j=1}^m z_{ij} w_j, i = 1, \dots, n \quad (3)$$

Remark. A benefit criterion means that the higher the value of the criterion the better the value is for a given alternative. For example, if a customer has a choice between two Internet Service Providers (ISPs) A [bandwidth = 500 Mbps, price = \$200] and B [bandwidth = 200 Mbps, price = \$50], and the criteria are price and bandwidth, then bandwidth is a benefit criterion since a customer would want higher bandwidths. On the other hand, price is a cost criterion since a customer would want to minimize the price he pays for the service.

III. THE M-AOMDV ROUTING PROTOCOL

Sections A-D describes the main components of the M-AOMDV routing protocol, while section E gives an illustrative example of how SAW works with the M-AOMDV routing protocol.

In a fully connected graph the number of paths between a given source and destination nodes could be very large. Finding the shortest path is NP-Hard [13] and in this work we follow the AOMDV approach by setting the number of discovered paths to 3 [23]. Hence, only 3 paths are stored in the mote routing table for a given destination route. The packet loss percentage metric is also stored at the motes routing table along with a SAW value, advertised hop count and last hop as shown in Table 2. The M-AOMDV has four phases which includes: route discovery (RD), packet loss discovery (PLD), route selection (RS); and, Route Maintenance (RM). Each phase is described in the following Sections.

Table 2. Columns in the Routing Table of M-AOMDV Protocol

| Measure | Value |
|----------------------|--------------|
| Dest. IP | 192.168.0.1 |
| Hop Count | 2 |
| Advertised Hop Count | 2 |
| Last Hop | 192.168.0.11 |
| Packet Loss | 0.20 |
| SAW Value | 0.33 |
| Next Hop | 192.168.0.2 |

A. Route Discovery Phase

The M-AOMDV routing uses the same method as the AOMDV protocol to discover and populate the motes routing for multiple routes to a given destination. During the protocol's route discovery phase, the packet loss percentage is set to zero and stored in each mote's routing table. Since at this stage there is no packet loss information, the shortest path is used for initial data transmission.

B. Packet Loss Discovery Phase

The source mote sends 5 collection probe packets (COLL packets) at 12 second intervals to the last hop address (stored in routing table) for each path to a given destination. The last hop mote receive the probe packets and waits on a timer to expire and then replys to the source mote with a reply COLL packet carrying one of these values 20%, 40%, 60%, 80% and 100%, based on how many probes it receives. If the source mote does not receive a reply from the last hop mote within a given time period, then the packet loss percentage is set to 100% for that path.

C. Route Selection Phase

After the PLD phase is completed multiple paths in a given routing table will have both hop count and packet loss information for a given destination mote. The information is then used by the SAW MCDM method describe in Section 2.2 to calculate a SAW value for each path. The shortest path selected at the RD phase is then replaced with the route to the destination with the highest SAW value.

D. Route Maintenance Phase

During normal network operations the packet loss percentage metric is periodically updated during route maintenance to ensure that paths selected for routing have the most recent metric values for the calculation of the SAW path value in the PLD phase. This phase ensures that packet losses experienced along a path, due to current packet dropping motes, are taken into consideration. Hence new SAW values could invoke alternate paths to be used to send packets to a given destination. If all paths to a given destination go down then the M-AOMDV routing protocol initiates a new RD phase.

E. Illustrative example of SAW with the M-AOMDV routing protocol

In this section we will present an illustrative example of how the M-AOMDV routing protocol uses SAW method. A WSN topology consisting of four node disjoint paths is shown in Fig. 3.

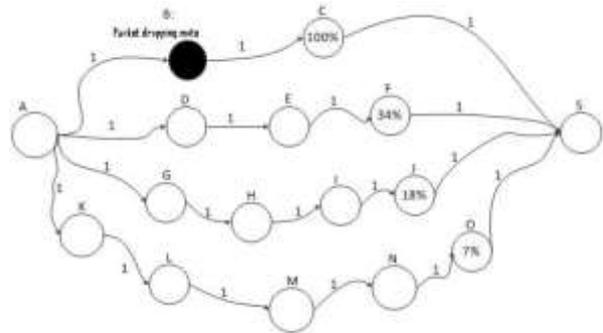


Fig. 3. WSN last mote values when the network is under attack/lossy conditions

The motes are labelled from A to O and S. The mote labelled A is the source and the node labelled S is the WSN sink. In this example, mote A chooses from four paths when it wants to send data to S. The packet loss percentage value for each path is indicated at the center of the last hop mote. The hop count for every edge is 1. The hop count is labelled on each link, while the packet loss percentage is labelled within the last hop motes on each path to the destination (S). Assume that data packets are currently being transmitted along path A-B-C-S. The packet dropping occurs at mote B. Assume that the percentage loss along this path is 100%. The mote C indicates this value, and mote A will record this percentage based on a timeout mechanisms for path A-B-C-S. The last hop motes (F, J, O) on paths A-D-E-F-S, A-G-H-I-J-S and A-K-L-M-N-O sends their packet loss percentages (34%, 18%, 7%) to the source mote.

The SAW method makes a multi-criteria path selection from four paths (see Fig. 1). SAW builds a WSN hop count and packet loss percentage function for each metric from the Eqs. 2.1 and 2.2. The routing metrics in Table 3 gives the SAW routing values.

Table 3. SAW metric values.

| Path | Hop count | Loss % | SAW Hop Count | SAW Loss % | Final SAW Hop Count and Loss % |
|---------------------|-----------|--------|---------------|------------|--------------------------------|
| A, B, C, S | 3.00 | 100.00 | 1.00 | 0.00 | |
| A, D, E, F, S | 4.00 | 34.00 | 0.67 | 0.71 | 0.40 |
| A, G, H, I, J, S | 5.00 | 18.00 | 0.33 | 0.88 | 0.69 |
| A, K, L, M, N, O, S | 6.00 | 7.00 | 0.00 | 1.00 | 0.66 |
| a | 3.00 | 7.00 | 1.00 | 1.00 | 0.60 |
| b | 6.00 | 100.00 | 0.00 | 0.00 | |
| weights | 0.40 | 0.60 | 0.40 | 0.60 | |

Table 4. ns2 [32] simulation settings

| Method | Description | Value |
|-------------------------|--|-------------------------------------|
| Simulator | Network Simulator version | ns2.34 |
| Channel type | Type of wireless channel | Channel/WirelessChannel |
| Radio-propagation model | Wireless propagation model | Propagation/ TwoRayGround |
| Network interface type | Type of physical interfaces | Phy/WirelessPhy |
| MAC type | The MAC layer | Mac/802.11 |
| Link layer type | The link layer | Link Layer (LL) |
| Antenna | Type of Antenna | Antenna/ OmniAntenna |
| ifqType | Type of Queue | Queue/DropTail/ priQueue |
| ifqLen | Length of the Queue | 50 |
| Area (mxm) | Size of simulation field | 800 x 800 |
| Motes | Number of WSN motes | 150 |
| Routing protocol | The routing protocol | AOMDV, M-AOMDV |
| Energy model | WSN energy model | EnergyModel |
| Encoding codec | Rate at which a codec's output data should be consumed | Constant Bit Rate (CBR) |
| Data frame length | Frame size is measured in bytes and has a minimum and maximum length | 50 bytes |
| Transport protocol | Open Systems Interconnection (OSI) transport-layer protocol | Transmission Control Protocol (TCP) |

IV. EXPERIMENTAL DESIGN

In this work an improvement to the AOMDV protocol is implemented so that it can react to lossy environments. Therefore, it is only necessary to compare the M-AOMDV protocol with the existing AOMDV protocol to show that the improved protocol is able to provide Fault tolerance to applications in the presence of lossy conditions. Cases of the two protocols with and without any packet dropping motes is given to illustrate that only in the case of lossy networks the protocols are different. The ns-2 simulator is used to evaluate the M-AOMDV routing protocol in WSNs. The selected simulation settings are shown in Table 4.

Simulations use the 'Loss- Monitor' objects, which implements a traffic sink and maintains statistics about the data it receives, for example, the number of packets lost and the amount of bytes received. These statistics are used to calculate the packet loss rate.

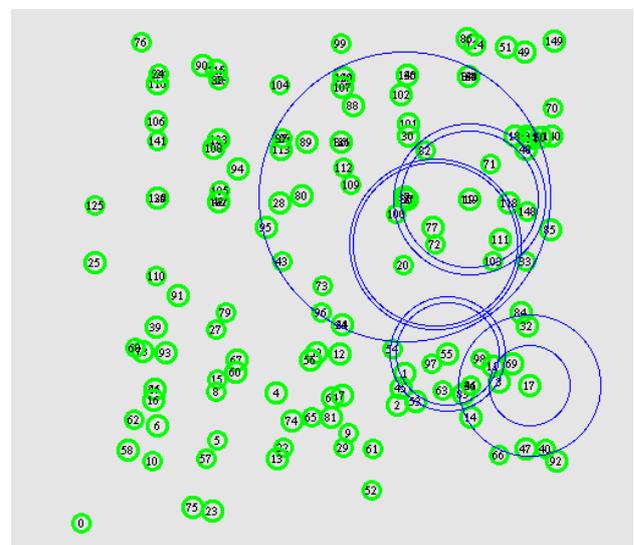


Fig. 4. Simulation WSN topology

The simulation consists of a WSN of 150 motes shown in see Fig. 4. The sensing motes (mote 52 through to

mote 75) are closer to the lower part and the sink, mote 99 is at the top. To simulate dangerous network conditions, where packets would be lost, we used packet dropping on seven network motes. These 'packet dropping' motes are closer to the sink mote, for example, mote 88. Seven different motes are attacked in each experimental run. There are 5 source motes (mote 0, mote 52, mote 92, mote 75 and mote 66) and each transmits a packet at Constant Bit Rate (CBR) over Transmission Control Protocol (TCP) every 0.015 seconds to the sink (mote 99). Sampling is done every 10 seconds with a total simulation time of 300 seconds.

A total of 100 runs are done under the same experimental conditions with the only variable being the seven packet dropping motes. The action of packet

dropping is initiated at 120 seconds for all runs, that is, the packet dropping commenced at this time.

V. EXPERIMENTAL RESULTS

The simulation results on packet loss rates are presented in this Section. A graph is displayed, which shows the performance statistics of the AOMDV and M-AOMDV routing protocols. Fig. 3 shows that the WSN with the AOMDV and M-AOMDV routing protocols with no packet dropping motes perform similarly, with a relatively stable packet loss rate of approximately 600 packets.

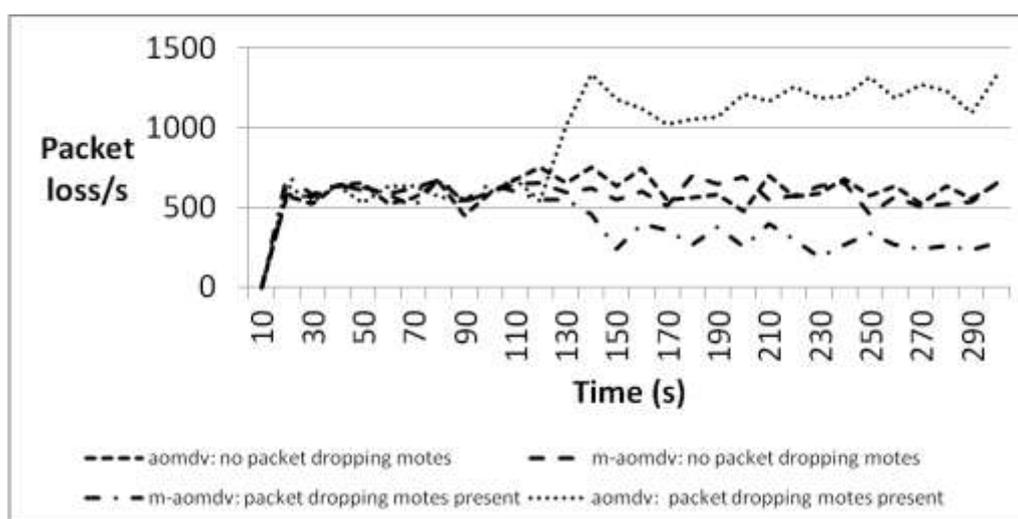


Fig. 3. Packet loss rates: AOMDV and M-AOMDV routing protocols

The WSN with the AOMDV routing protocol is very susceptible to lossy environments as it shows very high packet loss rates with an average of approximately 1150 packets. In comparison the WSN using the M-AOMDV routing protocol has a low average packet loss rate of roughly 300 packets. This indicates that the WSN with the M-AOMDV routing protocol provides greater Fault tolerance than with the AOMDV routing protocol and surprising better than without packet dropping.

When the WSN network is experiencing lossy conditions, the M-AOMDV routing protocol, switches to paths which are not congested. This is because it bases its path selection decisions on packet loss and hop count with a greater bias on packet loss. Thus, the routing algorithm chooses paths with fewer packet losses and these paths are longer. Initially, on these longer paths there is fresh buffer space on each mote. Generally during communication on longer paths there is a more balanced distribution of packets sent from mote to mote. Hence, buffers do not overflow as often as it does on shorter paths. Consequently, on longer paths there would be less overflowing of mote buffers. This would result in less dropped packets. The resulting fault tolerance to the sink mote would increase as more data is transmitted per unit time from source to destination motes.

VI. CONCLUSION

An extension to the AOMDV routing protocol for WSN is evaluated. It aims to improve the Fault tolerance of the AOMDV routing protocol in lossy environments. It is called the M-AOMDV routing protocol and operates in a WSN environment along node disjoint paths. An additional packet loss percentage metric is added to the WSN motes routing table. This M-AOMDV routing protocol is compared to the AOMDV routing protocol. Simulations using ns2 shows that the M-AOMDV routing protocol reduces packet loss rate in lossy conditions when compared with AOMDV. The protocol avoids paths that experiences very high packet drop rates.

REFERENCES

- [1] Abolhasan, M., Wysocki, T., & Dutkiewicz, E. (2004). A review of routing protocols for mobile ad hoc networks. *Ad hoc networks*, 2(1), 1-22.
- [2] Ahmadi, S. M. (2012). Analysis towards Mobile IPv4 and Mobile IPv6 in Computer Networks. *International Journal of Intelligent Systems and Applications (IJISA)*, 4(4), 33.
- [3] Al-Rodhaan, M., Mackenzie, L., & Ould-Khaoua, M. (2007). A new route discovery algorithm for MANETs

- with chase packets. *International Journal of Simulation Systems, Science & Technology, Special Issue on: Performance Modelling of Computer Networks, Systems and Services*, 8, 1-12.
- [4] Ammari, H. M., & Das, S. K. (2009). Fault tolerance measures for large-scale wireless sensor networks. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 4(1), 2.
- [5] Bansal, M., Rajput, R., & Gupta, G. (1999). Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations. *The Internet Society*.
- [6] Broch, J., Maltz, D. A., Johnson, D. B., Hu, Y. C., & Jetcheva, J. (1998, October). A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking* (pp. 85-97). ACM.
- [7] Bryson, N., & Mobolurin, A. (1997). An action learning evaluation procedure for multiple criteria decision making problems. *European Journal of Operational Research*, 96(2), 379-386.
- [8] De Couto, D. S., Aguayo, D., Bicket, J., & Morris, R. (2005). A high-throughput path metric for multi-hop wireless routing. *Wireless Networks*, 11(4), 419-434.
- [9] Demirbas, M. (2004). *Scalable design of fault-tolerance for wireless sensor networks* (Doctoral dissertation, The Ohio State University).
- [10] Draves, R., Padhye, J., & Zill, B. (2004, August). Comparison of routing metrics for static multi-hop wireless networks. In *ACM SIGCOMM Computer Communication Review* (Vol. 34, No. 4, pp. 133-144). ACM.
- [11] Draves, R., Padhye, J., & Zill, B. (2004, September). Routing in multi-radio, multi-hop wireless mesh networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking* (pp. 114-128). ACM.
- [12] Ganesan, D., Govindan, R., Shenker, S., & Estrin, D. (2001). Highly-resilient, energy-efficient multipath routing in wireless sensor networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(4), 11-25.
- [13] Garey, M. R., Johnson, D. S., & Tarjan, R. E. (1976). The planar Hamiltonian circuit problem is NP-complete. *SIAM Journal on Computing*, 5(4), 704-714.
- [14] Grefenstette, J. J. (Ed.). (2014). *Proceedings of the First International Conference on Genetic Algorithms and their Applications*. Psychology Press.
- [15] Guo-Yin, W. A. N. G. (2003). Calculation Methods for Core Attributes of Decision Table [J]. *Chinese Journal of Computers*, 5, 611-615.
- [16] Handziski, V., Köpke, A., Frank, C., & Karl, H. (2003). *Semantic addressing for wireless sensor networks*.
- [17] Koushanfar, F., Potkonjak, M., & Sangiovanni-Vincentelli, A. (2002). Fault tolerance techniques for wireless ad hoc sensor networks. In *Sensors, 2002. Proceedings of IEEE* (Vol. 2, pp. 1491-1496). IEEE.
- [18] Kumar, S., Bhardwaj, M., & Bhat, A. Q. (2013, February). Study of Wireless Sensor Networks its Routing Challenges and Available Sensor Nodes. In *International Journal of Engineering Research and Technology* (Vol. 2, No. 2 (February-2013)). ESRSA Publications.
- [19] Lee, S. J., & Gerla, M. (2001). Split multipath routing with maximally disjoint paths in ad hoc networks. In *Communications, 2001. ICC 2001. IEEE International Conference on* (Vol. 10, pp. 3201-3205). IEEE.
- [20] Marina, M. K., & Das, S. R. (2002). Ad hoc on-demand multipath distance vector routing. *ACM SIGMOBILE Mobile Computing and Communications Review*, 6(3), 92-93.
- [21] Metri, R., & Agrawal, S. (2014, April). Ant colony optimization algorithm based an intelligent protocol to improve QoS of MANETs. In *Circuits, Systems, Communication and Information Technology Applications (CSCITA), 2014 International Conference on* (pp. 121-125). IEEE.
- [22] Mueller, S., Tsang, R. P., & Ghosal, D. (2004). Multipath routing in mobile ad hoc networks: Issues and challenges. In *Performance tools and applications to networked systems* (pp. 209-234). Springer Berlin Heidelberg.
- [23] Nasipuri, A., & Das, S. R. (1999). On-demand multipath routing for mobile ad hoc networks. In *Computer Communications and Networks, 1999. Proceedings. Eight International Conference on* (pp. 64-70). IEEE.
- [24] Nasipuri, A., Castañeda, R., & Das, S. R. (2001). Performance of multipath routing for on-demand protocols in mobile ad hoc networks. *Mobile Networks and applications*, 6(4), 339-349.
- [25] Park, V. D., & Corson, M. S. (1997, April). A highly adaptive distributed routing algorithm for mobile wireless networks. In *INFOCOM'97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution., Proceedings IEEE* (Vol. 3, pp. 1405-1413). IEEE.
- [26] Perkins, C. E., & Royer, E. M. (1999, February). Ad-hoc on-demand distance vector routing. In *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA'99. Second IEEE Workshop on* (pp. 90-100). IEEE.
- [27] Perkins, C. E., & Royer, E. M. (1999, February). Ad-hoc on-demand distance vector routing. In *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA'99. Second IEEE Workshop on* (pp. 90-100). IEEE.
- [28] Raghavendra, C. S., Sivalingam, K. M., & Znati, T. (Eds.). (2004). *Wireless sensor networks*. Springer.
- [29] Raghavendran, C. V., Satish, G. N., & Varma, P. S. (2012). Intelligent Routing Techniques for Mobile Ad hoc Networks using Swarm Intelligence. *International Journal of Intelligent Systems and Applications (IJISA)*, 5(1), 81.
- [30] Rea, S., & Pesch, D. (2004, September). Multi-metric routing decisions for ad hoc networks using fuzzy logic. In *Wireless Communication Systems, 2004, 1st International Symposium on* (pp. 403-407). IEEE.
- [31] Saleh, I., El-Sayed, H., & Eltoweissy, M. (2006, November). A fault tolerance management framework for wireless sensor networks. In *Innovations in Information Technology, 2006* (pp. 1-5). IEEE.
- [32] Simulator, N. (1989). ns-2.
- [33] Tahir, M. A., Jamshed, A., Daadaa, Y., & Rehman, H. U. (2014, January). Dynamic QoS multicasting using greedy approach and fuzzy-based Tabu Search. In *Computer Applications and Information Systems (WCCAIS), 2014 World Congress on* (pp. 1-6). IEEE.
- [34] Van Mieghem, P., & Kuipers, F. A. (2004). Concepts of exact QoS routing algorithms. *Networking, IEEE/ACM Transactions on*, 12(5), 851-864.
- [35] Wang, T. Y., Han, Y. S., Varshney, P. K., & Chen, P. N. (2005). Distributed fault-tolerant classification in wireless sensor networks. *Selected Areas in Communications, IEEE Journal on*, 23(4), 724-734.
- [36] Wei, L., Yong-Feng, C., & Ya, L. (2014). An enhanced safety algorithm for network QoS multicast routing optimization.
- [37] Yang, K. (2014). *Wireless Sensor Networks*. Springer.
- [38] Yin, P. Y., Chang, R. I., Chao, C. C., & Chu, Y. T. (2014). Niche ant colony optimization with colony guides for QoS multicast routing. *Journal of Network and Computer Applications*, 40, 61-72.

- [39] Yuan, Y., Chen, H., & Jia, M. (2005, October). An optimized ad-hoc on-demand multipath distance vector (AOMDV) routing protocol. In *Communications, 2005 Asia-Pacific Conference on* (pp. 569-573). IEEE.
- [40] Zavadskas, E. K., Turskis, Z., Dejus, T., & Viteikiene, M. (2007). Sensitivity analysis of a simple additive weight method. *International Journal of Management and Decision Making*, 8(5), 555-574.

Authors' Profiles



Koffka Khan did his MSc and MPhil at The University Of The West Indies. He was awarded by the University of the West Indies for his contributions made in postgraduate work in 2009 as a research assistant. He is presently a student at The University of The West Indies; St. Augustine Campus (TRINIDAD & TOBAGO) in the Department of

Computing and Information Technology (Faculty of Science & Agriculture). He has up-to-date, published twelve papers in journals of international repute & in fourteen proceedings of international conferences.



Wayne Goodridge is a Lecturer in the Department of Computing and Information Technology, The University of the West Indies, St. Augustine. He did his PhD at Dalhousie University and his research interest includes computer communications and security.

How to cite this paper: Koffka Khan, Wayne Goodridge, "Fault Tolerant Multi-Criteria Multi-Path Routing in Wireless Sensor Networks", *International Journal of Intelligent Systems and Applications (IJISA)*, vol.7, no.6, pp.55-63, 2015. DOI: 10.5815/ijisa.2015.06.06